Web Appendix for

The Challenges of Deploying an AI Pricing Tool: Evidence from Airbnb

Section 1. Data Sources and Construction of the Data

1.1. Data Sources

We briefly discuss the main data sources that we have used in our paper and the variables that we obtained from each data source.

(i) InsideAirbnb.com: This is a publicly available website from which we got the list of all Airbnb listings in each of the 4 cities (Austin, Los Angeles, New York, San Francisco and Seattle) and the unique listing IDs associated with each of the listings. As we will discuss in section 1.2, we used this list to randomly sample the hosts that we study in our analysis.

(ii) AirDNA: We used AirDNA data to obtain information on daily prices, daily booking status, and other characteristics mentioned on webpages of each property. This data spans from October 2014 to Aug 2017, and it covers all Airbnb listings in the 4 cities.

(iii) Scraping: We scraped the websites of all properties in our random sample to get daily information on whether the host used Smart pricing algorithm, Custom price or Base price.

1.2. Sample Construction

We first obtained the data from AirDNA, which spans from October 2014 to Aug 2017, and it covers all

Airbnb listings in 4 cities (Austin, Los Angeles, San Francisco and Seattle). This data has information on daily prices, daily occupancy status, and all other characteristics mentioned on webpages of the properties.

Our next task was to randomly sample a set of properties from the entire set of Airbnb properties. To do so, we obtained a list of all Airbnb listings in each of the four cities from a publicly available website, viz., InsideAirbnb.com. There was a total of 29,315 properties in this list as of October 2014, and each listing was associated with a unique ID. We randomly selected 8,000 properties (roughly 27% of the listings IDs) from the overall list. We used Python's library, '*random.sample,'* to shuffle and randomly sample the 8,000 properties (roughly 27% of the listings IDs) from the overall list. The library *random.sample*(*population, k*) takes two inputs: *population* is the list or sequence from which the a random sample is chosen, and *k* is the length of returned sample list, that is, the number of random elements to choose from *population*. This random sampling was done without replacement. Calling *random.sample*(*population, k*) returns a new list containing *k* elements, which are randomly sampled from *population*. In our implementation of the library *random.sample*(*population, k*), we set *population* = the list of 29,315 property IDs and set *k* = 8000. The implementation thus returned a list of *k* = 8000 randomly selected property IDs.

Next, for each randomly selected property, we scraped its website to get its property calendar at the end of each month starting from Oct 2014. From the calendar pages that we scraped from the 8,000 listings every month, we obtained the information whether the property's price in each date was Base, Custom or Smart price. We then matched the listing of these 8,000 properties to the properties in the AirDNA data. Our final step was to deal with the 'stale vacancies' issue. A stale vacancy issue refers to a scenario where a property is listed but the host neglected to update the listing status. As a result, a property may appear to be available but would never be reserved because hosts did not respond to any booking request. Fradkin (2017) found that about 15% of the time that guest requests were rejected is due to this issue. To address this issue, following Zalmanson et al. (2018) we removed properties that did not have any booking one year prior to the natural experiment and throughout the whole observation window. This resulted in 6,678 properties, which represented our final sample.

Section 2. Reduced-form Regression Results of Other Cities Comparing Custom, Base, and Smart Prices

In this section, we present the results of the reduced-form analyses for the remaining cities other than Austin. Recall that we ran the following regressions in the main text:

$$Rev_{jt} = Constant + Custom_{jt} \times Month_t + Listing_j + Date_t + \varepsilon_{jt} \quad (1)$$

$$Rev_{jt} = Const + \alpha_1 Base_{jt} \times Month_t + \alpha_2 Custom_{jt} \times Month_t + Listing_j + Date_t + \varepsilon_{jt} \quad (2)$$

where j represents the listing and t is the calendar date. $Listing_j$ represents the listing j fixed effect and $Date_t$ are the date fixed effects. $Base_{jt}$ and $Custom_{jt}$ are binary variables that show whether host j chooses Base and Custom prices at night t (rather than Smart price). When running the regression in equation (1), we only consider observations before the introduction of SP (pre-SP), from Oct 2014 to Nov 2015, but when running regression (2), we use all observations in the data from Oct 2014 to Aug 2017. The DV for these regressions is the daily revenue earned by the host. As discussed in the main paper, to alleviate the selection problem, we use IPTW to match the treatment and control groups on observable characteristics.

The estimation results for Los Angeles, San Francisco, and Seattle are respectively presented in Tables A.1, A.2, and A.3. Column 1 in these tables shows the results for regression (1), which corresponds to the results of section 3.1 of the main paper. It suggests that Custom prices yield higher revenue/profits for hosts than Base prices. Columns 2 in these tables presents the results for regression (2), which corresponds to the results of section 3.2 of the main paper. It suggests that Smart prices yield higher revenue/profits for hosts than Base prices but lower revenue/profits than Custom prices.

Table A.1: Performance of Base and Custom prices as compared to Smart prices across different months in Los Angeles

VARIABLES	(1) Revenue	VARIABLES	(2) Revenue		(2) Revenue
Custom × Month 1	31.51***	Base × Month 1	-3.427	Custom × Month 1	8.305
	(2.833)		(3.898)		(5.307)
Custom × Month 2	28.18***	Base × Month 2	-2.922	Custom × Month 2	7.198*
	(2.884)		(3.827)		(4.000)
Custom × Month 3	27.74***	Base × Month 3	-5.579	Custom × Month 3	4.261
	(2.650)		(3.592)		(3.726)
Custom × Month 4	28.50***	Base × Month 4	-6.265*	Custom × Month 4	-0.593
	(2.691)		(3.225)		(3.132)
Custom × Month 5	24.09***	Base × Month 5	-6.854**	Custom × Month 5	-1.203
	(2.700)		(3.050)		(3.004)
Custom × Month 6	27.09***	Base × Month 6	-6.946**	Custom × Month 6	4.194
	(2.728)		(3.153)		(3.539)
Custom × Month 7	24.07***	Base × Month 7	-3.471	Custom × Month 7	8.031**
	(2.788)		(3.422)		(3.706)
Custom × Month 8	24.42***	Base × Month 8	-5.512	Custom × Month 8	4.256
	(3.058)		(3.388)		(3.122)
Custom × Month 9	8.837***	Base × Month 9	3.051	Custom × Month 9	4.114
	(2.303)		(3.147)		(3.418)
Custom × Month 10	17.58***	Base × Month 10	-0.908	Custom × Month 10	8.511**
	(2.062)		(3.461)		(3.606)
Custom × Month 11	16.43***	Base × Month 11	-5.425*	Custom × Month 11	2.836
	(2.106)		(3.119)		(3.210)
Custom × Month 12	32.86***	Base × Month 12	-3.496	Custom × Month 12	5.062
	(3.568)		(2.945)		(3.216)
Constant	48.68***			Constant	64.43***
	(0.764)				(1.195)
Observations	525,675	Observations	1,033,183		1,033,183
R-squared	0.400	R-squared	0.435		0.435
Matched Sample	YES	Matched Sample	YES		YES
Property FE	YES	Listing FE	YES		YES
Date FE	YES	Date FE	YES		YES

Clustered Robust standard errors in parentheses *** p<0.01, ** p<0.05, * p<0.1

Table A.2: Performance of Base and Custom prices as compared to Smart prices across different months in San Francisco

VARIABLES	(1) Revenue	VARIABLES	(2) Revenue		(2) Revenue
Custom × Month 1	54.45***	Base × Month 1	-3.104	Custom × Month 1	13.18**
	(4.997)		(6.470)		(6.569)
Custom × Month 2	54.08***	Base × Month 2	-10.26*	Custom × Month 2	3.842
	(4.449)		(6.132)		(6.147)
Custom × Month 3	45.74***	Base × Month 3	-18.50***	Custom × Month 3	-0.665
	(4.636)		(6.222)		(6.662)
Custom × Month 4	42.94***	Base × Month 4	-12.67**	Custom × Month 4	4.375
	(4.506)		(4.914)		(4.755)
Custom × Month 5	43.04***	Base × Month 5	-10.98**	Custom × Month 5	6.469
	(4.461)		(4.363)		(4.664)
Custom × Month 6	32.25***	Base × Month 6	-2.888	Custom × Month 6	9.976**
	(4.724)		(4.569)		(4.586)
Custom × Month 7	29.59***	Base × Month 7	-2.599	Custom × Month 7	9.092
	(4.992)		(4.906)		(6.678)
Custom × Month 8	26.15***	Base × Month 8	-5.862	Custom × Month 8	8.027
	(4.787)		(5.626)		(5.433)
Custom × Month 9	29.86***	Base × Month 9	-10.74	Custom × Month 9	-0.831
	(4.427)		(6.855)		(7.305)
Custom × Month 10	34.61***	Base × Month 10	-6.309	Custom × Month 10	18.31***
	(3.360)		(5.378)		(5.273)
Custom × Month 11	18.59***	Base × Month 11	-10.31*	Custom × Month 11	-1.559
	(3.227)		(5.513)		(5.122)
Custom × Month 12	53.25***	Base × Month 12	-12.76**	Custom × Month 12	1.953
	(4.486)		(5.145)		(5.240)
Constant	69.31***			Constant	99.16***
	(1.540)				(1.920)
Observations	336,482	Observations	691,660		691,660

R-squared	0.371	R-squared	0.402		0.402
Matched Sample	YES	Matched Sample	YES		YES
Property FE	YES	Listing FE	YES		YES
Date FE	YES	Date FE	YES		YES

Clustered Robust standard errors in parentheses *** p<0.01, ** p<0.05, * p<0.1

Table A.3: Performance of Base and Custom prices as compared to Smart prices across different months in Seattle

VARIABLES	(1) Revenue	VARIABLES	(2) Revenue		(2) Revenue
Custom × Month 1	14.48***	Base × Month 1	-4.187	Custom × Month 1	-3.182
	(4.239)		(4.091)		(3.786)
Custom × Month 2	15.13***	Base × Month 2	-5.163	Custom × Month 2	-4.147
	(3.726)		(4.807)		(3.929)
Custom × Month 3	17.32***	Base × Month 3	-1.523	Custom × Month 3	1.127
	(3.729)		(5.015)		(5.001)
Custom × Month 4	11.17***	Base × Month 4	1.753	Custom × Month 4	-0.00707
	(3.807)		(4.905)		(4.992)
Custom × Month 5	17.72***	Base × Month 5	-4.075	Custom × Month 5	2.018
	(4.061)		(4.256)		(4.410)
Custom × Month 6	8.120*	Base × Month 6	2.532	Custom × Month 6	13.70***
	(4.171)		(4.819)		(4.683)
Custom × Month 7	13.64***	Base × Month 7	-1.048	Custom × Month 7	16.60***
	(4.148)		(5.553)		(4.984)
Custom × Month 8	10.35**	Base × Month 8	2.199	Custom × Month 8	13.74***
	(4.098)		(4.817)		(4.853)
Custom × Month 9	12.98***	Base × Month 9	-10.58*	Custom × Month 9	-0.782
	(3.323)		(5.486)		(4.987)
Custom × Month 10	16.40***	Base × Month 10	-9.307**	Custom × Month 10	0.439
	(3.079)		(4.549)		(4.266)
Custom × Month 11	10.72***	Base × Month 11	0.718	Custom × Month 11	6.015
	(2.947)		(4.697)		(4.318)
Custom × Month 12	20.65***	Base × Month 12	3.970	Custom × Month 12	8.640*
	(4.098)		(3.896)		(4.675)
Constant	51.40***			Constant	61.61***
	(1.191)				(1.461)
Observations	165,978	Observations	352,820		352,820
R-squared	0.363	R-squared	0.377		0.377
Matched Sample	YES	Matched Sample	YES		YES
Property FE	YES	Listing FE	YES		YES
Date FE	YES	Date FE	YES		YES

Clustered Robust standard errors in parentheses *** p<0.01, ** p<0.05, * p<0.1

Section 3. Details of the IPTW method and the Conditional c- Dependence test

Details of IPTW: We dealt with the self-selection issue in our reduced-form analysis in sections 3.1 and 3.2 of the main paper with the Inverse Probability of Treatment Weighting (IPTW) method. IPTW is a weighting method to construct a balanced sample of treated and untreated units (Rosenbaum 1987, Austin and Stuart 2015). This method first computes the weight for each given unit as the inverse of its treatment probability, and then weighs all units to eliminate any existing systematic differences between treated and untreated units in terms of the observed covariates that explain the treatment probability. The treatment probability (also called propensity score) is the probability that an individual unit is assigned to the treatment condition, conditional on a set of observed variables (Rosenbaum and Rubin 1983). Unlike matching methods in which units in the two groups that are not very similar to each other are discarded, the IPTW method does not have to discard any unit when creating a balanced sample (Guo and Fraser 2015). Prior Monte Carlo studies have shown that IPTW method leads to lower mean squared error in the estimates of treatment effects as compared to those obtained from matching methods such as Propensity Score Matching (Austin 2013, Austin and Stuart 2015). In what follows, we explain how we implement the IPTW for regressions (1) and (2) in sections 3.1 and 3.2 of the main paper.

To implement IPTW for regression (1) in section 3.1, where the data from before the introduction of SP is used and a host has only two pricing options, Base (control) or Custom (treatment), we first estimate the propensity score for each unit i as a function of a set of observed covariates X_i, $\widehat{ps}_i = f(X_i\beta)$, which impact host i's decision of using Custom versus Base price (or assignment into the treatment condition). We approximate propensity score $\widehat{ps}_i$ by estimating the parameter vector β by fitting a logistic regression where the input is the vector of observed covariates X_i and the output is a binary response, which equals 1 if unit i was observed to receive treatment (i.e., uses Custom price) and equals 0 if otherwise (i.e., uses Base price). Following Zhang et al. (2021), we include covariates that capture characteristics of the host and property to estimate the treatment probabilities in the first stage of IPTW: number of bedrooms, number of bathrooms, property type (apartment/house), entire home, Superhost, number of properties, respond always, respond fast, host since, number of reviews, number of photos, max guests, minimum stay, instant book enabled, cleaning fee, business ready, cancellation policy, have extra people fee, have deposit, and pre-SP custom price usage. These covariates are also correlated with the DV in the subsequent regression (1). The estimation process finds a parameter vector β that maximizes the data likelihood of observed treatment assignments (Rosenbaum and Rubin 1983), where f(.) takes a logit functional form. Given the estimate of β, we compute each unit i's propensity score as $\widehat{ps}_i = f(X_i\beta)$, which we use to compute the weights of the sub-sample of the observations of each property i as (T is the binary treatment indicator, which takes the value of 1 if the unit i belongs to the treatment group) $w_i = \frac{T}{ps_i} + \frac{1-T}{1-ps_i}$. These weights capture the contribution of different units when estimating the average treatment effects in the subsequent regression in equation (1) of the paper. In other words, we estimate the average treatment effects in the subsequent DiD regression in equation (1) of the paper using weighted least squares, with w_i being the weight of each unit i. By weighting each unit i's observations by w_i, IPTW creates a 'synthetic sample' of weighted treatment and control groups in which the covariates X are balanced across the two weighted groups. Since the covariates X also include the potential observed confounders, the covariate balance across the weighted treatment and control groups ensures that the treatment assignment across the two weighted groups is independent of any observed confounder; further if there were no unobserved confounders, this would in turn imply that the weighted treatment and control groups are as good as data being sampled from a population in which treatment assignments were random.

Our next step is to validate whether the resulting treated and control groups (in the weighted sample) are comparable in terms of observed covariates. Table A.4 compares the balance of observed covariates across control and treated groups before and after weighting for observations used in regression in equation (1) of the paper. The results suggest that IPTW method weights treated and control groups so that the observed covariates are similar across the groups (we can reject the null hypothesis that the control and treated observations are similar before weighting at the 5% significance level, $pvalue < 0.001$, but we cannot reject the null after weighting, $pvalue > 0.05$). This indicates that our IPTW strategy created a weighted sample that effectively removed significant imbalances in the observed covariates that may have existed in the raw sample.

Table A.4: Comparing control (Base) and treatment (Custom) observations in pre-SP periods before and after IPTW

	Before Weighting		After Weighting			Before Weighting		After Weighting	
Variable	Control	Treated	Control	Treated	Variable	Control	Treated	Control	Treated
Bedrooms	1.31	1.26	1.30	1.26	No. of	16.41	19.13	17.22	18.37
	(0.88)	(0.81)	(0.87)	(0.80)	photos	(11.88)	(12.93)	(12.28)	(12.40)
Bathrooms	1.26	1.20	1.24	1.22	Max guests	3.22	3.33	3.27	3.26
	(0.71)	(0.52)	(0.66)	(0.59)		(2.26)	(2.16)	(2.25)	(2.13)
Apartment	0.60	0.69	0.63	0.65	Minimum	4.00	4.35	4.02	4.53
	(0.49)	(0.46)	(0.48)	(0.48)	stay	(14.90)	(20.07)	(14.85)	(22.78)
House	0.31	0.24	0.29	0.27	Instant book	0.13	0.20	0.15	0.18
	(0.46)	(0.42)	(0.45)	(0.44)		(0.34)	(0.40)	(0.36)	(0.39)
Entire	0.54	0.61	0.56	0.57	Cleaning fee	53.12	60.93	55.36	57.92
	(0.50)	(0.49)	(0.50)	(0.49)		(55.40)	(51.61)	(54.53)	(51.72)
Superhost	0.19	0.23	0.21	0.22	Business	0.15	0.22	0.18	0.20
	(0.40)	(0.42)	(0.41)	(0.41)	ready	(0.36)	(0.42)	(0.38)	(0.40)

No. of properties	3.45 (8.06)	6.85 (23.63)	4.32 (18.40)	5.44 (18.36)	Cancellation	2.31 (0.79)	2.54 (0.67)	2.39 (0.76)	2.46 (0.71)
Respond always	0.66 (0.47)	0.72 (0.45)	0.69 (0.46)	0.70 (0.46)	Have extra fee	0.53 (0.50)	0.60 (0.49)	0.55 (0.50)	0.59 (0.49)
Respond fast	0.38 (0.48)	0.48 (0.50)	0.41 (0.49)	0.45 (0.50)	Have deposit	0.57 (0.50)	0.62 (0.48)	0.58 (0.49)	0.61 (0.49)
Host since	-1041.25 (511.89)	-1043.41 (509.37)	-1043.97 (512.19)	-1038.77 (511.23)	Custom use pre SP	0.21 (0.24)	0.76 (0.26)	0.31 (0.29)	0.62 (0.32)
No. of reviews	46.50 (62.14)	66.83 (68.95)	54.10 (67.16)	61.23 (65.19)					

To implement IPTW for regression in equation (2) in section 3.2 of the paper, where the data from before and after the introduction of SP is used and a host has three pricing options after SP, Base (control) or Custom (treatment 1) or Smart (treatment 2), we can use a similar method as above but we need to account for multiple treatments. To address the multiple treatments, we use the multinomial logit in the first step of IPTW, i.e., we estimate the propensity score of receiving treatment t for each unit i as a function of a set of observed covariates X_i, $\widehat{ps}_{it} = f_t(X_i\beta)$, which impact host i's decision of using Custom/Base/Smart price, by fitting a multinomial logistic regression where the input is the vector of observed covariates X_i and the output takes the value of 1 if unit i was observed to receive treatment 1 (i.e., uses Custom price), 2 if it was observed to receive treatment 2 (i.e., uses Smart price), and equals 0 if otherwise (i.e., uses Base price). Given the estimate of β, we compute each unit i's propensity score to receive treatment t as $\widehat{ps}_{it} = f_t(X_i\beta)$, which we use to compute the weights of the sub-sample of the observations of each property i as (T is the treatment indicator, which takes the value of 1 if the unit i belongs to the treatment group 1, 2 if it belongs to the treatment group 2, and 0 otherwise) $w_i = \frac{1\{T=0\}}{ps_{i0}} + \frac{1\{T=1\}}{ps_{i1}} + \frac{1\{T=2\}}{ps_{i2}}$. Table A.5 compares the balance of observed covariates across control and treated groups before and after weighting for observations after the introduction of SP used in regression in equation (2) of the paper. The results suggest that IPTW method weights treated and control groups so that the observed covariates are similar across the groups (we can reject the null hypothesis that the control and treated observations are similar before weighting at the 5% significance level, $pvalue < 0.001$, but we cannot reject the null after weighting, $pvalue > 0.05$).

Table A.5 Comparing control (Base), treatment 1 (Custom), and treatment 2 (Smart) observations in post-SP periods before and after IPTW

	Before Weighting			After Weighting		
Variable	Control	Treatment 1	Treatment 2	Control	Treatment 1	Treatment 2
Bedrooms	1.32 (0.91)	1.34 (0.88)	1.28 (0.83)	1.31 (0.88)	1.33 (0.89)	1.30 (0.85)
Bathrooms	1.28 (0.75)	1.24 (0.55)	1.21 (0.51)	1.25 (0.64)	1.25 (0.58)	1.23 (0.56)
Apartment	0.55 (0.50)	0.65 (0.48)	0.59 (0.49)	0.61 (0.49)	0.61 (0.49)	0.61 (0.49)
House	0.35 (0.48)	0.26 (0.44)	0.30 (0.46)	0.30 (0.46)	0.29 (0.46)	0.30 (0.46)
Entire	0.53 (0.50)	0.64 (0.48)	0.54 (0.50)	0.58 (0.49)	0.59 (0.49)	0.58 (0.49)
Superhost	0.22 (0.42)	0.30 (0.46)	0.31 (0.46)	0.25 (0.43)	0.27 (0.44)	0.25 (0.43)
No. of properties	3.96 (10.92)	8.06 (26.77)	3.61 (5.89)	9.00 (45.96)	5.26 (17.52)	3.99 (7.76)
Respond always	0.69 (0.46)	0.79 (0.41)	0.78 (0.41)	0.73 (0.44)	0.76 (0.43)	0.74 (0.44)
Respond fast	0.39 (0.49)	0.54 (0.50)	0.58 (0.49)	0.44 (0.50)	0.46 (0.50)	0.46 (0.50)
Host since	-1021.59 (537.02)	-1049.58 (531.84)	-1050.12 (538.56)	-1030.23 (543.21)	-1044.46 (540.03)	-1041.68 (540.03)
No. of reviews	50.48 (65.28)	78.71 (72.96)	81.12 (73.32)	62.39 (73.96)	64.97 (67.40)	64.98 (64.12)
No. of photos	16.93 (12.20)	19.98 (12.74)	20.33 (13.95)	18.26 (12.51)	18.49 (12.09)	18.63 (12.29)
Max guests	3.26	3.56	3.32	3.32	3.38	3.32

	(2.33)	(2.30)	(2.18)	(2.29)	(2.25)	(2.18)
Minimum stay	4.11 (17.38)	3.91 (11.92)	4.91 (36.83)	4.61 (19.91)	4.21 (14.18)	4.23 (18.12)
Instant book	0.14 (0.34)	0.22 (0.42)	0.27 (0.44)	0.18 (0.38)	0.18 (0.39)	0.19 (0.39)
Cleaning fee	53.61 (56.27)	64.18 (53.54)	60.94 (52.13)	59.21 (55.62)	59.17 (53.66)	60.15 (52.38)
Business ready	0.14 (0.34)	0.23 (0.42)	0.20 (0.40)	0.17 (0.38)	0.18 (0.38)	0.18 (0.39)
Cancellation	2.33 (0.79)	2.59 (0.63)	2.52 (0.67)	2.46 (0.73)	2.47 (0.70)	2.46 (0.71)
Have extra fee	0.56 (0.50)	0.61 (0.49)	0.71 (0.45)	0.60 (0.49)	0.59 (0.49)	0.61 (0.49)
Have deposit	0.57 (0.50)	0.63 (0.48)	0.65 (0.48)	0.60 (0.49)	0.60 (0.49)	0.61 (0.49)
Custom use pre SP	0.31 (0.32)	0.61 (0.34)	0.55 (0.35)	0.54 (0.37)	0.37 (0.35)	0.46 (0.37)

The unbiasedness of our reduced form results relies on the conditional independence assumption (CIA) in our IPTW analysis. The CIA assumption is that conditional on the distributions of the observed covariates f(X), the potential outcome Y is independent of the treatment assignment T. This is because the treatment probabilities (propensity scores) are computed as a function of observables. Although in the above tables we have shown that the weighted sample of treated units and control units are comparable in terms of the observed covariates, it is still possible that there were relevant variables omitted from the IPTW implementation. If there are unobservables that affect hosts' pricing method decision, then this may introduce a bias in the estimated treatment effect as the unobservables could be correlated with the dependent variable in the subsequent regressions in equations (1) and (2) of the main paper. To deal with the issue of unobserved confounders, researchers have used the conditional c-dependence sensitivity test proposed by Masten and Poirier (2018) to assess the robustness of the average treatment effect to unobserved confounders (Zhang et al. 2021). We discuss this as follows.

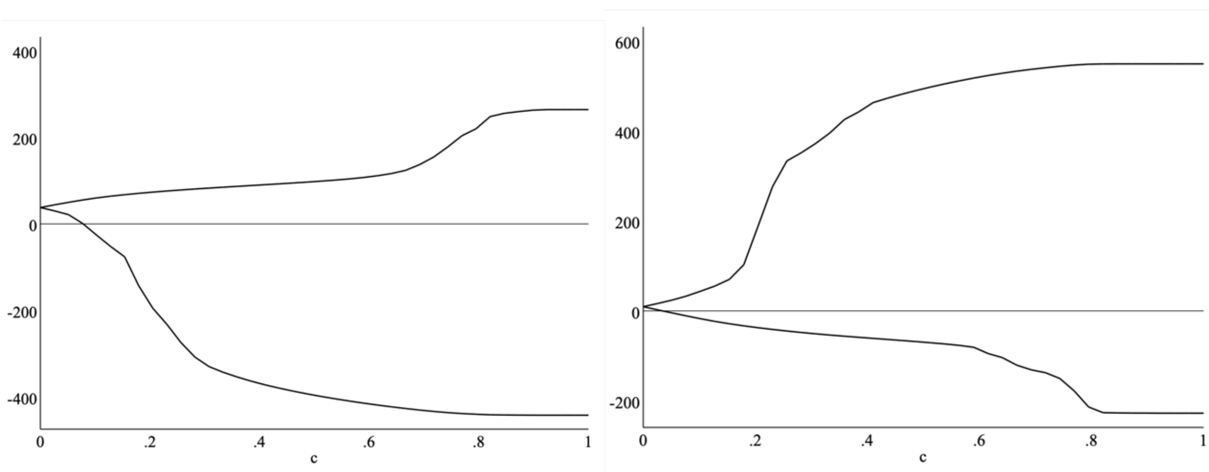
<u>Conditional c-Dependence Test for Unobserved Confounders:</u> The objective of the conditional c-dependence exercise is to inform us how large the impact of hypothetical unobserved confounders needs to be in order for the treatment effect to be nullified – the larger the value of unobserved confounders that is needed to nullify the treatment effect, the greater will be the confidence that our estimated treatment effect will be robust to unobserved confounders.

To see how conditional c-dependence analysis works, first note that conditional independence assumption (CIA) means that the probability of being selected for treatment conditional on the observed covariates, X, and the unobserved potential outcome variable is equal to the probability of treatment conditional on the observed covariates X only. The conditional c-dependence assumption states that under a hypothetical scenario in which CIA is violated, the two aforementioned conditional probabilities would not be identical, but should be within a distance from each other. Under conditional c-dependence, the deviation from CIA, as captured by the maximum distance between the two aforementioned conditional probabilities, is described by a scalar variable c. If CIA were true, the two conditional probabilities would be equal to each other, and the value of c would be zero. If CIA is violated, there will be a non-zero distance between the two conditional probabilities, bounded by the value of c. This distance c captures the impact of unobserved confounders. The greater the magnitude of c, the greater will be the impact of unobserved confounders, and consequently, the greater be violation of CIA.

Given these primitives, the objective of the conditional c-dependence exercise is to calculate the minimum amount of deviation required from the CIA (or the minimum value of c) that would nullify the treatment effect that we had estimated assuming CIA to hold true. Intuitively, if it requires a large value of c to nullify a treatment effect obtained under CIA (i.e., the hypothetical unobserved factors need to be so large enough to violate CIA), then it would imply that the estimate of the treatment effect that we had obtained by assuming CIA is quite robust to hypothetical unobserved confounders. On the other hand, if a small value of c can nullify the treatment effect, then it would suggest that the estimate of the treatment effect that we had obtained under CIA is sensitive to the hypothetical unobserved confounders. This minimum value of c required to nullify the treatment effect is called 'breakdown value'. For more details of the c-dependence sensitivity analysis see Zhang et al (2021).

The result of the c-dependence sensitivity analysis for regressions (1) of section 3.1 in the main text is presented in panel (a) of Figure A.1. The breakdown point is at c = 0.078. To understand what this value implies in terms of percentage changes in treatment probabilities, note that the mean treatment (Custom price usage) probability in pre-SP periods is 44%. Thus if we were to translate our result in terms of percentage changes in Custom usage probabilities, our estimated breakdown points imply that in order for our estimated average treatment effect of the Custom pricing to be nullified, the potential unobservables that enter the host's decision of using Custom pricing must be large enough so that the unobservables cause a deviation of at least 0.078/0.44 = 17.7% change in the treatment probability (i.e., Custom price usage). This is comparable to the robustness measures reported in prior literature (Zhang et al., 2021; Masten and Poirier, 2018).

Similarly, the c-dependence sensitivity analysis for regressions (2) of section 3.2 in the main text is presented in panel (b) of Figure A.1. Here, the breakdown point is at c = 0.034. The mean treatment probability for Smart (Custom) price in periods after SP in our sample is 7% (44%). Thus, our estimated breakdown point implies that in order for our estimated average treatment effect of the Smart (Custom) pricing to be nullified, the potential unobservables that enter the host's decision of using Smart (Custom) pricing must be large enough so that the unobservables cause a deviation of at least 0.034/0.07 = 48.6% (0.034/0.44 = 7.7%) change in the Smart (Custom) price usage probability. These values are comparable to the robustness measures reported in prior literature (Zhang et al., 2021; Masten and Poirier, 2018).



(a) Conditional c-dependence bounds for regression (1) (b) Conditional c-dependence bounds for regression (2)

Figure A.1: Conditional c-dependence sensitivity analysis for the reduced form results presented in the main paper

Section 4. Testing for the Parallel Trends Assumption

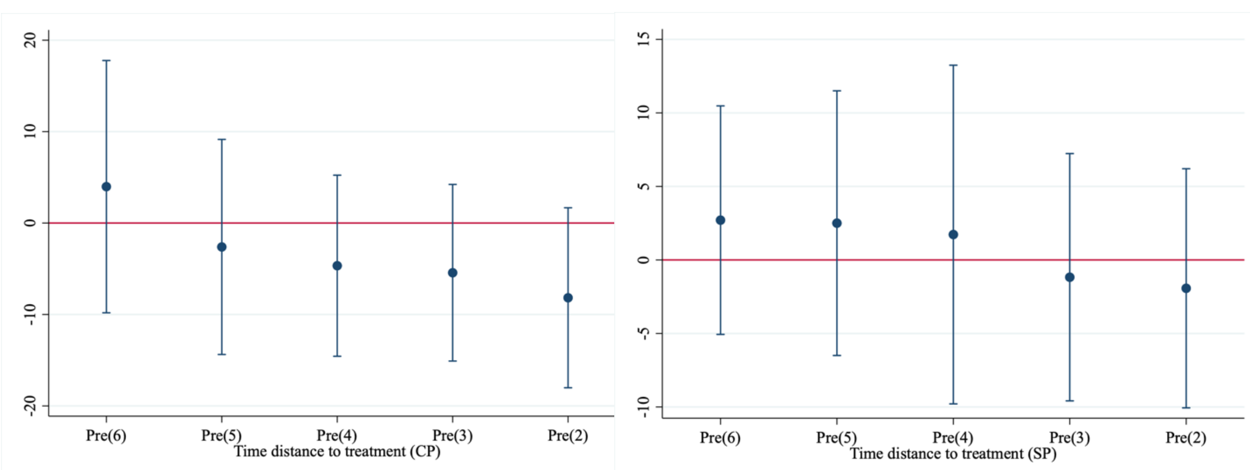
In regression in equation (1) of section 3.1 and regression in equation (2) of section 3.2 in the paper, we used the standard two-way fixed effects difference-in-differences (TWFE-DiD) along with the IPTW method to study the revenue of different pricing methods (Base/Custom/Smart). Here, we examine the pre-treatment trends to ensure that the weighted sample of treatment and control groups followed similar trends in revenues. The standard approach of doing so is by estimating a relative-time model (Autor 2003), which decomposes the pre-treatment periods in terms of a series of period dummies and estimates the coefficients of all those dummies that are within k periods prior to the treatment. To conduct this analysis for regression in equation (1) in section 3.1 of the paper, we estimate the following equation for the city of Austin using the weights that we got from the IPTW method:

$$Rev_{jt} = Cons + Custom_{jt} \times Month_t + \sum_{i=1}^{+6} \beta_i PreCP(i)_{jt} \times CP\_Treated_j + Listing_j + Date_t + \varepsilon_{jt} \quad \text{(A.1)}$$

and to conduct this analysis for regression in equation (2) in section 3.2 of the paper, we run the following equation for the city of Austin using the weights that we got from the IPTW method:

$$Rev_{jt} = Cons + \alpha_1 Base_{jt} \times Month_t + \alpha_2 Custom_{jt} \times Month_t + \sum_{i=1}^{+6} \beta_i PreSP(i)_{jt} \times SP\_Treated_j + Listing_j + Date_t + \varepsilon_{jt} \quad \text{(A.1)}$$

In equations (A.1) and (A.2), j represents the listing identity and t is the calendar date. $Base_{jt}$ and $Custom_{jt}$ are binary variables that show whether host j chooses Base and Custom prices at night t (rather than Smart price). $CP\_Treated_j$ and $SP\_Treated_j$ are binary variables that respectively represent whether host j ever used Custom and Smart pricing during our data sample. The series of time dummies $PreCP(i)_{jt}$ equals 1 if period t is i months prior to the month in which property i used Custom pricing for the first time, and equals 0 otherwise. Similarly, the series of time dummies $PreSP(i)_{jt}$ equals 1 if period t is i months prior to the month in which property i used SP for the first time, and equals 0 otherwise. The parameters $\{\beta_i\}$ in equation (A.1) identify the trend in the dependent variable for the pre-treatment periods in the treatment group that uses Custom price relative to the control group that uses Base price. Similarly, the parameters $\{\beta_i\}$ in equation (A.2) identify the trend in the dependent variable for the pre-treatment periods in the treatment group that uses Smart price relative to the remaining groups. In both regressions in equations (A.1) and (A.2), we set the period prior to the treatment month as reference period (i.e. $\{\beta_i\}_{i=1}$ was normalized to zero). When running the regression in equation (A.1), we only consider observations before the introduction of SP (pre-SP), from Oct 2014 to Nov 2015, but when running regression in equation (A.2), we use all observations in the data from Oct 2014 to Aug 2017. The estimated parameters $\{\beta_i\}$ in equations (A.1) and (A.2) are respectively presented in panels (a) and (b) of Figure A.2. As we can see, the estimated coefficients for the pre-treatment dummies are all statistically insignificant at the 5% level. Thus, the parallel trends assumption cannot be rejected.



(a) Estimated coefficients β_i for regression (A.1) (b) Estimated coefficients β_i for regression (A.2)

Figure A.2: Similarity of pre-treatment trends between treated and control groups for the reduced form results of the main paper

Section 5. Hosts' Learning about Demand Shifters/Competitive Behavior over Time

Here, we regress Custom prices in the city of Austin on the interaction of month fixed effects with the variable *Experienced*, which equals one if the host has more than two years of experience on Airbnb and zero otherwise, controlling for listing fixed effects and time fixed effects. The results are presented in column 1 of Table A.7. They demonstrate that there is no statistically significant difference in monthly variation of Custom prices when the host is experienced versus inexperienced. The variation of Custom price level across months is related to how host incorporates changes in demand and supply conditions in her prices. So, column 1 in Table A.6 implies that it is unlikely that host learns about demand shifters or the competitors' behavior as she gains more experience on the platform.

Table A.6: Seasonal variation in Custom price levels in Austin when the host is experienced versus inexperienced (column 1) and when the host has multiple versus single listing (column 2)

	(1)		(2)
VARIABLES	Custom Price	VARIABLES	Custom Price
Experienced × Month 1	1.066	Pro × Month 1	13.22*
	(8.190)		(7.089)
Experienced × Month 2	3.154	Pro × Month 2	21.82*
	(13.35)		(12.66)
Experienced × Month 3	26.03	Pro × Month 3	19.07
	(17.12)		(20.91)
Experienced × Month 4	4.607	Pro × Month 4	17.36*
	(11.35)		(9.724)
Experienced × Month 5	9.780	Pro × Month 5	16.94*
	(9.299)		(9.163)
Experienced × Month 6	13.34	Pro × Month 6	14.67
	(11.21)		(10.40)
Experienced × Month 7	13.78	Pro × Month 7	18.39*
	(11.02)		(10.76)
Experienced × Month 8	18.22*	Pro × Month 8	20.10*
	(11.05)		(10.61)
Experienced × Month 9	8.371	Pro × Month 9	15.21
	(10.09)		(10.02)
Experienced × Month 10	12.86	Pro × Month 10	10.66
	(8.731)		(10.38)
Experienced × Month 11	6.186	Pro × Month 11	13.81
	(10.34)		(8.432)
Experienced × Month 12	5.491	Pro × Month 12	-
	(10.68)		-
Constant	255.4***	Constant	253.6***
	(4.324)		(4.580)
Observations	119,686	Observations	119,686
R-squared	0.828	R-squared	0.827
Matched Sample	YES	Matched Sample	YES
Listing, Date FE	YES	Listing, Date FE	YES

Clustered Robust standard errors in parentheses *** p<0.01, ** p<0.05, * p<0.1

Section 6. Differences between Professional and Non-Professional Hosts in Custom Pricing Behavior

In this section, we regress Custom prices in the city of Austin on the interaction of month fixed effects with the variable *Pro,* which equals one if the host has more than one listing on Airbnb (i.e., the host is professional) and zero otherwise. The results of this analysis are shown in column 2 of Table A.6. This result shows that the variation of Custom prices across months, after controlling for listing and time fixed effects, is not significantly different across professionals who have multiple listings and non-professionals who have only one listing. Assuming that professionals set profit-maximizing Custom prices, this result suggests that non-professionals also care about maximizing profits

Section 7. The effect of competitors on the focal host's reservation

In this section, we examine the effect of the prices of competing properties in the same neighborhood on the occupancy of a focal property. To investigate this, we run the following regressions using the data from Austin:

$$Booked_{jt} = \alpha \log\left(Price_{jt}\right) + \beta \sum_{k \neq j} \log\left(Price_{kt}\right) + Listing_j + Date_t + \varepsilon_{jt} \tag{A.1}$$

Where $Booked_{jt}$ represents whether or not listing j was booked at time t, $Price_{jt}$is the price of listing j at time t, and $Price_{kt}$ is the price of competitor listing k which is in the same neighborhood as listing j, and the remaining variables are listing and date fixed effects. The estimated value for coefficient β is -0.0057 (p-value = 0.355) and for coefficient α is 0.018 (p-value = 0.024). This result suggests that while the own price coefficient is statistically significant, the prices of competing properties in the same neighborhood are not correlated with the occupancy of the focal property.

Section 8. When does higher revenue imply higher profit? A sensitivity analysis

In the main text, we compare the performance of Base, Custom, and Smart prices in terms of revenue. The results imply the following inequalities at the (average) monthly level:

$$\text{Revenue from Custom price} \geq \text{Revenue from Smart price} \geq \text{Revenue from Base price} \quad \text{(A.4)}$$

The question is whether these inequalities hold for profits too. Since marginal costs are not directly observed in the data, it is challenging to evaluate profits. We tackle this challenge by running a sensitivity analysis in which we assume different values for marginal cost and compare the profits associated with Base, Custom, and Smart prices conditional on the assumed values for marginal cost. This analysis will show us whether the inequalities in (A.4) hold for profits, given the assumed values of marginal cost, and how sensitive they are to the assumed value of marginal cost.

To conduct the sensitivity analysis, we use the data from Austin city and assume a fixed ratio for host's marginal cost to pre-SP average price (e.g., 0.1, 0.2, 0.3, etc). Given the assumed ratio, we then compute profits of each host j at every night t as follows:

$$Profit_{jt} = (price_{jt} - MC_j) \cdot D_{jt} = (price_{jt} - \gamma \cdot (preSP\ AvgPrice_j)) \cdot D_{jt} \quad \text{(A.5)}$$

where D_{jt} is a dummy variable that represents whether or not listing j is booked at night t, MC_j is the marginal cost, and parameter γ is the assumed fixed ratio of marginal cost to pre-SP average price. Finally, we run the following regression given the assumed value of γ:

$$Profit_{jt} = Custom_{jt} + Base_{jt} + Listing_j + Date_t + \varepsilon_{jt} \quad \text{(A.6)}$$

We vary γ between 0.1 to 0.6 and present the estimation results using the data from Austin in Table A.7. This table demonstrates that the inequalities "Profit of Custom price ≥ Profit of Smart price ≥ Profit of Base price" hold true up to $\gamma = 0.5$. At very high values of $\gamma \geq 0.55$, the second inequality flips so that the average profit from the Base price is higher than that of the Smart price. Note that we compare the average profits across Base, Custom, and Smart prices in here. This is unlike the regressions in sections 3.1 and 3.2 where we compared the profits at the monthly levels. The results are similar even if we compare the profits at the monthly level.

The above discussion suggests that the comparison in (A.4) between the performance of Base, Custom, and Smart prices which were based on the reduced-form analysis of revenue, can be extended to the analysis of profit so long as the host's marginal cost is not too high. In our structural estimations, the pre-SP average price charged by hosts is \$319.17 and the estimated average marginal cost across hosts is \$62, which imply a ratio of marginal cost to pre-SP average price of 0.194. This ratio is well below the threshold 0.55 required for the robustness of our reduced-form results.

Table A.7: Sensitivity of the comparison between profits of Base, Custom, and Smart prices to marginal cost in Austin city

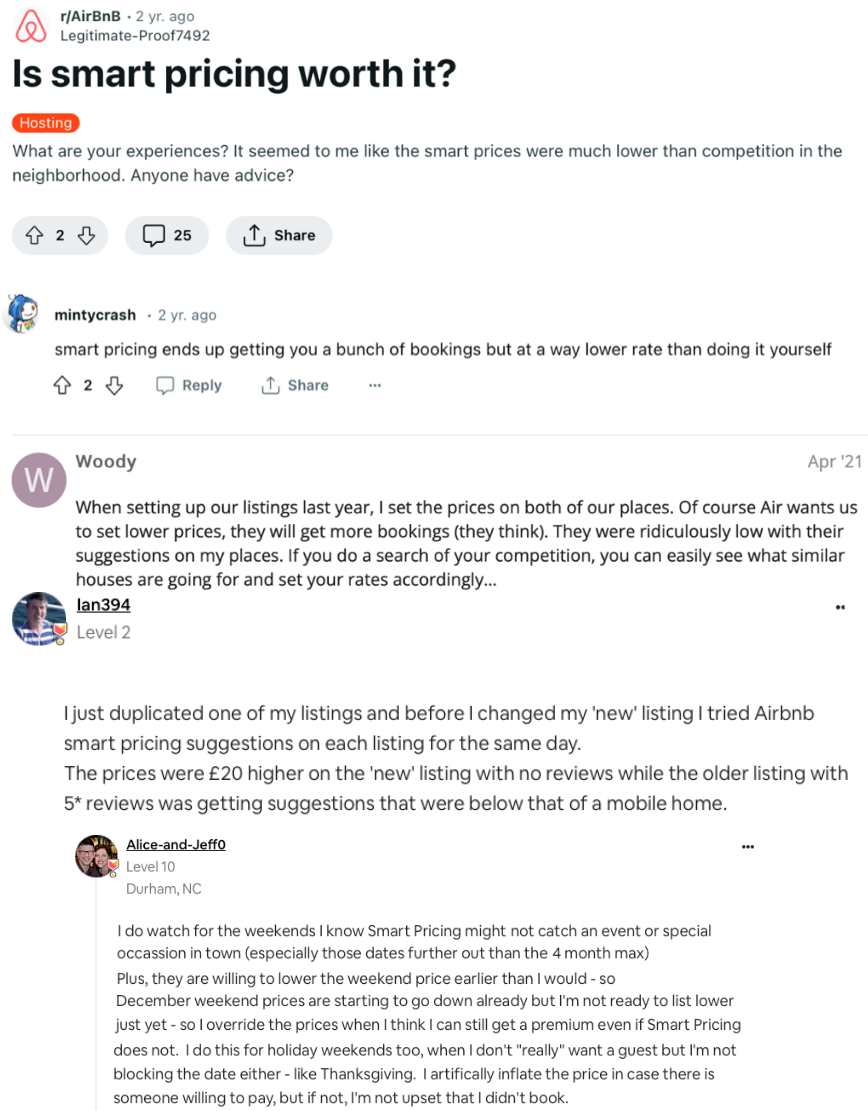
VARIABLES	(1) Profit ($\gamma = 0.1$)	(2) Profit ($\gamma = 0.2$)	(3) Profit ($\gamma = 0.3$)	(4) Profit ($\gamma = 0.4$)	(5) Profit ($\gamma = 0.5$)	(6) Profit ($\gamma = 0.55$)	(7) Profit ($\gamma = 0.6$)
Base	-2.579	-1.106	0.368	1.841	3.314	4.050*	4.787**
	(3.114)	(2.796)	(2.516)	(2.288)	(2.127)	(2.078)	(2.051)
Custom	6.298**	6.809**	7.320***	7.831***	8.342***	8.597***	8.853***
	(3.159)	(2.836)	(2.550)	(2.317)	(2.152)	(2.100)	(2.072)
Constant	46.45***	40.37***	34.30***	28.23***	22.16***	19.12***	16.09***
	(2.068)	(1.855)	(1.671)	(1.524)	(1.428)	(1.402)	(1.392)
Observations	809,227	809,227	809,227	809,227	809,227	809,227	809,227

R-squared	0.296	0.292	0.285	0.276	0.263	0.254	0.245
Matched Sample	YES	YES	YES	YES	YES	YES	YES
Property FE	YES	YES	YES	YES	YES	YES	YES
Date FE	YES	YES	YES	YES	YES	YES	YES

Robust standard errors in parentheses *** p<0.01, ** p<0.05, * p<0.1

Section 9. Anecdotal Evidence of Hosts' Perceptions about SP on Online Airbnb Host Forums

Figure A.3 presents a few examples from online forums in which Airbnb hosts talk about their concerns about the SP algorithm. Another example can be found at https://airhostsforum.com/t/smart-pricing-needs-to-be-smarter/27926/3.


r/AirBnB · 2 yr. ago
Legitimate-Proof7492

Is smart pricing worth it?

Hosting

What are your experiences? It seemed to me like the smart prices were much lower than competition in the neighborhood. Anyone have advice?

2 25 Share

mintycrash · 2 yr. ago

smart pricing ends up getting you a bunch of bookings but at a way lower rate than doing it yourself

2 Reply Share ···

Woody Apr '21

When setting up our listings last year, I set the prices on both of our places. Of course Air wants us to set lower prices, they will get more bookings (they think). They were ridiculously low with their suggestions on my places. If you do a search of your competition, you can easily see what similar houses are going for and set your rates accordingly...

Ian394
Level 2

I just duplicated one of my listings and before I changed my 'new' listing I tried Airbnb smart pricing suggestions on each listing for the same day.
The prices were £20 higher on the 'new' listing with no reviews while the older listing with 5* reviews was getting suggestions that were below that of a mobile home.

Alice-and-Jeff0
Level 10
Durham, NC

I do watch for the weekends I know Smart Pricing might not catch an event or special occassion in town (especially those dates further out than the 4 month max)
Plus, they are willing to lower the weekend price earlier than I would - so
December weekend prices are starting to go down already but I'm not ready to list lower just yet - so I override the prices when I think I can still get a premium even if Smart Pricing does not. I do this for holiday weekends too, when I don't "really" want a guest but I'm not blocking the date either - like Thanksgiving. I artifically inflate the price in case there is someone willing to pay, but if not, I'm not upset that I didn't book.

Figure A.3: Examples of hosts talking about SP on online forums

Section 10. Details of the Bayesian Estimation of Demand and Supply Models

We first describe the procedure for demand estimation. In Bayesian estimation using the MCMC method, one typically makes use of the prior distribution and the likelihood of the observed data to obtain a new draw from the posterior distribution of parameters. However, when directly drawing from the posterior distribution is difficult, draws from the posteriors can be implemented using a Metropolis-Hasting algorithm. For the estimation purposes, we partition demand parameters into two parts: 1) the vector of parameters associated with the utility coefficients, $\theta_u = (\bar{\alpha}, \bar{\gamma}, \Sigma_\gamma)$, and 2) the fixed effects and parameters associated with the control function, $\theta_f = (\lambda_1, \lambda_2, \lambda_3, \{Zipcode_j, Year_t, Month_t, Weekend_t\})$. Suppose that we are at iteration r. We denote candidate parameters with "star" superscript (e.g. θ^{*r}) and the accepted parameters with no specific superscript other than the iteration number (e.g. θ^r). The estimation proceeds as follows:

1. Draw candidate parameters $\theta_u^{*r} = (\bar{\alpha}^{*r}, \bar{\gamma}^{*r}, \Sigma_\gamma^{*r})$ from a proposal distribution, $q(\theta_u^{*r} | \theta_u^{r-1})$. Throughout, unless explicitly explained, we use a Random-Walk (RW) Metropolis chain for the proposal density. We impose further restrictions in the estimation process to make sure parameters lie within reasonable range. For example, to make sure that the candidate parameters for standard deviations are positive, whenever a negative draw is realized, we immediately throw away that draw and re-draw from the proposal distribution until we get a positive candidate draw. Then, compute the likelihood conditional on θ_u^{*r} and θ_f^{r-1}, $\mathcal{L}(data | \theta_u^{*r}, \theta_f^{r-1})$. We repeat the same step to obtain the likelihood conditional on θ_u^{r-1} and θ_f^{r-1}, $\mathcal{L}(data | \theta_u^{r-1}, \theta_f^{r-1})$. Then, we determine whether or not to accept θ_u^{*r} with the acceptance probability λ, where $\lambda = \min\{\frac{\mathcal{L}(data | \theta_u^{*r}, \theta_f^{r-1})\ q(\theta_u^{r-1} | \theta_u^{*r})}{\mathcal{L}(data | \theta_u^{r-1}, \theta_f^{r-1})\ q(\theta_u^{*r} | \theta_u^{r-1})}, 1\}$.
2. Draw candidate parameters θ_f^{*r} from a proposal distribution, $q(\theta_f^{*r} | \theta_f^{r-1})$. Then, compute the likelihood conditional on θ_f^{*r} and θ_u^r, $\mathcal{L}(data | \theta_u^r, \theta_f^{*r})$. We repeat the same step to obtain the likelihood conditional on θ_u^r and θ_f^{r-1}, $\mathcal{L}(data | \theta_u^r, \theta_f^{r-1})$. Then, we determine whether or not to accept θ_f^{*r} with the acceptance probability $\lambda = \min\{\frac{\mathcal{L}(data | \theta_u^r, \theta_f^{*r})\ q(\theta_f^{r-1} | \theta_f^{*r})}{\mathcal{L}(data | \theta_u^r, \theta_f^{r-1})\ q(\theta_f^{*r} | \theta_f^{r-1})}, 1\}$
3. Go to the next iteration (i.e., start from step 1 with $r + 1$).

Next, we explain the estimation of supply-side parameters. We estimate the value of σ_ω^2 from the observed Smart prices in the data and keep it fixed during the estimation of remaining parameters, $(\{c_j, \mu_j, AC_j, mc_j, FC_j\}_j, \sigma_\rho^2, \sigma_\eta)$, using the IJC method (Imai et al., 2009). Parameters that are specified at the level of each host are $(c_j, \mu_j, AC_j, mc_j, FC_j)$, and the parameters that are the same across all subjects are σ_ρ^2 and σ_η. The IJC method draws Markov chains from the posterior distribution of parameters. The structural estimation algorithm of a dynamic problem can typically be broken down into two components: outer loop and inner loop. In Bayesian estimation, the outer loop is implemented using the Markov Chain Monte Carlo (MCMC) method. The inner loop, which is nested within the outer loop, computes the value function and choice-specific value functions for constructing the likelihood. We now outline the IJC algorithm step-by-step. This method uses the Hierarchical Bayes approach to estimate individual-specific unobserved heterogeneity. Specifically, we impose a parametric assumption on the joint distribution of c_j, μ_j, mc_j, FC_j, and estimate the parameters of this distribution alongside the other parameters. So, the structural parameters being estimated for the supply side can be summarized by $(m_{c,\mu}, \Sigma_{c,\mu}, \{c_j, \mu_j, AC_j, mc_j, FC_j\}_j, \sigma_\rho, \sigma_\eta)$, where $m_{c,\mu}, \Sigma_{c,\mu}$ are the mean and variance-covariance matrix parameters explaining the joint distribution of costs and beliefs. To explain the estimation strategy, we partition the supply-side structural parameters into three parts: 1) the vector of hyperparameters $\theta_h = (m_{c,\mu}, \Sigma_{c,\mu})$ which determines the joint distribution of costs and mean belief across the population. 2) a set of vectors $\{\theta_j\}_j = \{(c_j, \mu_j, AC_j, mc_j, FC_j)'\}_j$ which summarizes the individual-specific parameters, and 3) $\theta_c = (\sigma_\rho, \sigma_\eta)$ which includes the parameters that are common across individuals. We will take the Hierarchical Bayes approach and treat $\{c_j, \mu_j, AC_j, mc_j, FC_j\}_j$ as individual-specific parameters in our estimation that capture unobserved heterogeneity.

Suppose that we are at iteration r. We start with $H^r = \{\theta_c^{*l}, \theta_{j_l}^{*l}, s^l, \widetilde{\mathcal{W}}^l(., s^l; \theta_{j_l}^{*l}, \theta_c^{*l})\}_{l=r-N}^{r-1}$, where j_l is the identity of the host whose pseudo-$E_\epsilon max$ function is randomly chosen to be stored at iteration l and N is the number of past iterations used for the expected value function approximation. The superscript l indicates that they are derived at outer loop iteration l. Since the IJC algorithm approximates the value function, we will use the terms, pseudo-value function ($\widetilde{\mathcal{W}}^l$), pseudo-expected value function ($\widetilde{\mathbb{E}}^l_{\mu'_j,s'}\,\mathcal{W}$), pseudo-alternative specific value function ($\widetilde{W}^l_a$) and pseudo-likelihood ($\tilde{L}^l$) when we outline the algorithm. The algorithm in each iteration r proceeds as follows:

1. Drawing hyperparameters (θ_h) step: Draw $m^r_{c,\mu}$ (population mean of θ_j) from the posterior density conditional on $\Sigma^{r-1}_{c,\mu}$ and $\{\theta^{r-1}_j\}_j$: $m^r_{c,\mu} \sim N(\frac{1}{J}\sum_{j=1}^J \theta^{r-1}_j, \frac{1}{J}\Sigma^{r-1}_{c,\mu})$. Then, draw $\Sigma^r_{c,\mu}$ (population var-cov matrix of θ_j) from the posterior density conditional on $m^r_{c,\mu}$ and $\{\theta^{r-1}_j\}_j$: $\Sigma^r_{c,\mu} \sim IW(\boldsymbol{C}^{-1}, J)$ where IW stands for the Inverse Wishart distribution and $\boldsymbol{C} = [\Theta^{r-1} - m^r_{c,\mu}][\Theta^{r-1} - m^r_{c,\mu}]^{\mathrm{T}}$ is a var-cov matrix calculated based on the last iteration individual-specific draws $\Theta^{r-1} = (\theta^{r-1}_1, \theta^{r-1}_2, \ldots, \theta^{r-1}_J)$ and the mean parameters drawn above. Note that this step does not involve the solution of the DP problem. As noted before, we summarize these two parameters as $\theta^r_h = (m^r_{c,\mu}, \Sigma^r_{c,\mu})$.
2. Data augmentation step: For each host j, we first draw a vector of candidate parameters $\theta^{*r}_j = (c^{*r}_j, \mu^{*r}_j, mc^{*r}_j, FC^{*r}_j)$ from the proposal density $\theta^{*r}_j \sim q(\theta^{*r}_j | \theta^{r-1}_j, \theta^r_h)$. The proposal density is a Random-Walk Metropolis chain that has a mean of θ^{r-1}_j and a variance proportional to the Cholesky decomposition of $\Sigma^r_{c,\mu}$ (Train, 2009). We impose further restrictions in the estimation process to make sure parameters lie within a reasonable range. For example, to make sure that the candidate parameters for costs are positive, whenever a negative draw is realized, we immediately throw away that draw and re-draw from the proposal distribution until we get a positive candidate draw. Then, compute the pseudo-likelihood for host j at θ^{*r}_j: $\tilde{L}^r_j(data_j | \theta^{*r}_j, \theta^{r-1}_c) = \prod_{t\in\{\text{Pre SP}\}} Pr_{jt}(a_{jt}|z_t; \theta^{*r}_j) Pr\,(p^{a_{jt}}_{jt}|z_t; \theta^{*r}_j) \prod_{t\in\{\text{Post SP}\}} \frac{\exp\left(\widetilde{W}^r_{a_{jt}}\left(on_{jt}, s_t; \theta^{*r}_j, \theta^{r-1}_c\right)\right)}{\sum_{a'} \exp\left(\widetilde{W}^r_{a'}\left(on_{jt}, s_t; \theta^{*r}_j, \theta^{r-1}_c\right)\right)}\, Pr\,(p^{a_{jt}}_{jt}|s_t; \theta^{*r}_j, \theta^{r-1}_c)$ where $data_j = \{on_{jt}, s_t, a_{jt}, p_{jt}\}_t$ includes observations associated with host j. The distribution of observed price, $p^{a_{jt}}_{jt}$, depends on the choice of price method, a_{jt}. If the choice of price method is Base or Smart, ($a_{jt} \in \{B, S\}$), the distribution of observed price is deterministic (to the econometrician). If the choice of price method is Custom ($a_{jt} = C$), the distribution of observed price is obtained from the first order condition $p^C_{jt} = mc_j - Q_{jt}(p^C_{jt})\left[\frac{\partial Q_{jt}(p = p^C_{jt})}{\partial p}\right]^{-1} + \eta_{jt}$. This distribution depends on the distribution of the error term, η_{jt}, by using change-of-variable calculus. Assuming that $\eta_{jt} \sim N(0, \sigma^2_\eta)$, where σ_η is a constant positive number, the distribution of observed price is of non-standard form because the above equation is implicit in price, i.e., price appears on both the left and right side of the equation. The distribution of observed prices is obtained by defining a new variable $r_{jt} = p_{jt} + f(p_{jt}) - mc_j$ that is distributed normal with mean 0 and variance σ^2_η. The likelihood for price is obtained in the standard way as the likelihood for r_{jt} multiplied by $|\frac{\partial r_{jt}}{\partial p_{jt}}| = |1 + \frac{\partial f}{\partial p_{jt}}|$. To obtain $\widetilde{W}^r_a$, we need to have $\widetilde{\mathbb{E}}^r_{s'}\,[\mathcal{W}(on'_j, s'; \theta^{*r}_j, \theta^{r-1}_c | on_j, s, a_j)$, which is obtained by a weighted average of past pseudo-$E_\epsilon max$ functions stored for approximation, $\{\widetilde{\mathcal{W}}^l(.,s^l; \theta^{*l}_{jl}, \theta^{*l}_c)\}_{l=r-N}^{r-1}$, treating θ^{*r}_j as one of parameters when computing the weights. In the case of independent kernels, for all on'_j, $\widetilde{\mathbb{E}}^r_{s'}\,[\mathcal{W}(on'_j, s'; \theta^{*r}_j, \theta^{r-1}_c | on_j, s, a_j) = \sum_{l=r-N}^{r-1} \widetilde{\mathcal{W}}^l(on'_j, s^l; \theta^{*l}_{jl}, \theta^{*l}_c)\frac{K_h(\theta^{*l}_c, \theta^{r-1}_c)\, K_h(\theta^{*l}_{jl}, \theta^{*r}_j)\, K_h(s^l, s')}{\sum_k K_h(\theta^{*k}_c, \theta^{r-1}_c)\, K_h(\theta^{*k}_{jk}, \theta^{*r}_j)\, K_h(s^k, s')}$. We repeat the same step to obtain the pseudo-likelihood conditional on $(\theta^{r-1}_j, \theta^{r-1}_c\})$, $\tilde{L}^r_j(data_j | \theta^{r-1}_j, \theta^{r-1}_c)$. Then, we determine whether or not to accept θ^{*r}_j with the acceptance

probability of $\lambda = \min\{\frac{\phi(\theta_j^{*r}|\theta_h^r)\,\tilde{L}_j^r(data_j|\theta_j^{*r},\theta_c^{r-1})\,q(\theta_j^{r-1}|\theta_j^{*r},\theta_h^r)}{\phi(\theta_j^{r-1}|\theta_h^r)\,\tilde{L}_j^r(data_j|\theta_j^{r-1},\theta_c^{r-1})\,q(\theta_j^{*r}|\theta_j^{r-1},\theta_h^r)}, 1\}$. Note that when θ_j^{*r} is accepted, it means the prior parameter μ_j^{*r} is accepted. This automatically implies the belief path resulting from μ_j^{*r} is also accepted.

3. Drawing common parameters: Draw candidate parameters $\theta_c^{*r} = (\sigma_\rho^{*r}, \sigma_\eta^{*r})$ from a proposal distribution, $q(\theta_c^{*r}|\theta_c^{r-1})$. Then, compute the pseudo-likelihood conditional on θ_c^{*r} and $\{\theta_j^r\}_j$: $\tilde{L}^r\left(data\middle|\{\theta_j^r\}_j, \theta_c^{*r}\right) =$

   $\prod_j\ [\prod_{t\in\{\text{Before SP}\}} Pr_{jt}(a_{jt}|z_t;\theta_j^r) Pr\ (p_{jt}^{a_{jt}}|z_t;\theta_j^r) \prod_{t\in\{\text{After SP}\}} \frac{\exp\left(\widetilde{W}_{a_{jt}}^r(on_{jt},s_t;\theta_j^r,\theta_c^{*r})\right)}{\sum_{a'}\exp\left(\widetilde{W}_{a'}^r(on_{jt},s_t;\theta_j^r,\theta_c^{*r})\right)}\ Pr\ (p_{jt}^{a_{jt}}|s_t;\theta_j^r,\theta_c^{*r})]$

   where $data = \{on_{jt}, s_t, a_{jt}, p_{jt}\}_{j,t}$ includes observations of all hosts. To obtain $\widetilde{W}_a^r$, we need to have $\widetilde{\mathbb{E}}_{s'}^r\ [\mathcal{W}(on_j', s'; \theta_j^r, \theta_c^{*r}|on_j, s, a_j)$, which is obtained by a weighted average of past pseudo-$E_\epsilon max$ functions stored for approximation, $\{\widetilde{\mathcal{W}}^l(.,s^l;\theta_{j_l}^{*l},\theta_c^{*l})\}_{l=r-N}^{r-1}$, treating θ_j^r as one of parameters when computing the weights. In the case of independent kernels, for all on_j', $\widetilde{\mathbb{E}}_{s'}^r\ [\mathcal{W}(on_j', s'; \theta_j^r, \theta_c^{*r}|on_j, s, a_j) = \sum_{l=r-N}^{r-1} \widetilde{\mathcal{W}}^l(on_j', s^l; \theta_{j_l}^{*l}, \theta_c^{*l})\frac{K_h(\theta_c^{*l},\theta_c^{*r})\,K_h\left(\theta_{j_l}^{*l},\theta_j^r\right)K_h(s^l,s')}{\sum_k K_h(\theta_c^{*k},\theta_c^{*r})\,K_h\left(\theta_{j_k}^{*k},\theta_j^r\right)K_h(s^k,s')}$. We repeat the same step to obtain the pseudo-likelihood conditional on $(\{\theta_j^r\}_j, \theta_c^{r-1}\})$, $\tilde{L}^r\left(data\middle|\{\theta_j^r\},\theta_c^{r-1}\right)$. Then, we determine whether or not to accept θ_c^{*r} with the acceptance probability of $\lambda = \min\{\frac{\tilde{L}^r(data|\{\theta_j^r\},\theta_c^{*r})\,q(\theta_c^{r-1}|\theta_c^{*r})}{\tilde{L}^r(data|\{\theta_j^r\},\theta_c^{r-1})\,q(\theta_c^{*r}|\theta_c^{r-1})}, 1\}$.

4. Randomly pick one host j, and choose one random period t from the observations belonging to host j_r. Denote $s^r = s_t$. Then, compute the pseudo-expected value function at $(\theta_{j_r}^{*r}, \theta_c^{*r})$, for all on_{j_r}', conditional on s^r: $\widetilde{\mathbb{E}}_{s'}^r\ [\mathcal{W}(on_j', s'; \theta_{j_r}^{*r}, \theta_c^{*r}|on_{j_r}, s^r, a_{j_r}) = \sum_{l=r-N}^{r-1} \widetilde{\mathcal{W}}^l(on_{j_r}', s^l; \theta_{j_l}^{*l}, \theta_c^{*l})\frac{K_h(\theta_c^{*l},\theta_c^{*r})\,K_h\left(\theta_{j_l}^{*l},\theta_{j_r}^{*r}\right)K_h(s^l,s')}{\sum_k K_h(\theta_c^{*k},\theta_c^{*r})\,K_h\left(\theta_{j_k}^{*k},\theta_{j_r}^{*r}\right)K_h(s^k,s')}$. Then, compute the pseudo-alternative specific value function, $\widetilde{W}_a^r(.,s^r;\theta_{j_r}^{*r},\theta_c^{*r})$ using pseudo-expected value function obtained from above. Finally, compute and store the pseudo-$E_\epsilon max$ function $\widetilde{\mathcal{W}}^r(.,s^r;\theta_{j_r}^{*r},\theta_c^{*r}) = \ln\left[\sum_a exp\left(\widetilde{W}_a^r(.,s^r;\theta_{j_r}^{*r},\theta_c^{*r})\right)\right]$

5. Update H^r to $H^{r+1}$and go to the next iteration (i.e., start from step 1 with $r + 1$).

Section 11. Estimates of the remaining demand-side and supply side structural parameters

Table A.8 presents the estimates of demand parameters that are not provided in the main text.

Table A.8: Estimated demand-side parameters using a subsample of data from Austin

	Parameter	Posterior Estimates: Mean	95% CI
Year FE	2015	-2.122	[-2.192, -2.054]
	2016	-2.535	[-2.662, -2.406]
Day FE	Weekday	-3.500	[-3.628, -3.346]
	Weekend	-2.633	[-2.772, -2.483]
	λ_1	1.060	[1.012, 1.110]
	λ_2	-0.725	[-0.781, -0.681]
	λ_3	-0.323	[-0.489, -0.206]

Note: The table presents posterior mean of parameters and their 95% credible intervals; values are computed using 20,000 draws with the first 15,000 draws removed.

Table A. presents the results of the OLS estimation of the parametric equation (18) for hosts' prior beliefs in the supply side model. Using these estimates, the estimated value for the standard deviation of error term in equation (18) is σ_ω= 0.3078.

Table A.9: SP belief regression coefficients

VARIABLES	log(Smart price)	VARIABLES	log(Smart price)
Intercept	-0.90	Superhost	-0.005
	(0.69)		(0.04)
Bedrooms	0.18***	No. of photos	-0.003***
	(0.03)		(0.0007)
Bathrooms	-0.11*	Overall rating	0.90***
	(0.04)		(0.10)
Apartment	0.45***	Max guests	0.10***
	(0.04)		(0.01)
House	0.23***	Security deposit	0.00002
	(0.04)		(0.00008)
Entire	0.25***	Cleaning fee	0.005***
	(0.04)		(0.0006)
No. of reviews	-0.06**	Min stay	0.01***
	(0.02)		(0.003)
R-squared		0.80	
No. of obs.		1,218	
Year, Month, Weekend FE		YES	

***p < 0.001; **p < 0.01; *p < 0.05

Section 12. Replication of the SP algorithm

In this section, we replicate the Smart Pricing algorithm following Ye et al. (2018). The step-by-step replication procedure is as follows. The first replication step is a binary classification model that predicts the booking probability of each listing-night. The classification is done by training a Gradient Boosting Machine, a non-linear mapping from a large set of raw features to an estimated booking probability. The features used to train this model include a variety of market information such as listing characteristics (number of bedrooms, property type, max guests, cleaning fee, number of reviews, etc), temporal features (listing's historical occupancy rate, blocking rate, etc), location features (zip code and neighborhood dummies), and supply/demand dynamics (day-of-week dummy, day-of-year dummy, year dummy, number of available listings in the neighborhood and in the city, etc). As standard in the literature, we split the data into three parts. The first part is used for training, the second part is used for testing, and the third part is a hold-out sample to validate the predictions of our replicated algorithm. The hold-out sample includes the observations that are used in the structural estimation. This is to ensure that the algorithm is not overfitted to these observations which are important for counterfactual simulations. The split between the train set (65%) and test set (35%) is random. The accuracy of our trained model in this step on the test set is 91.82%. We plot the most important features of the trained model in Figure A.4.

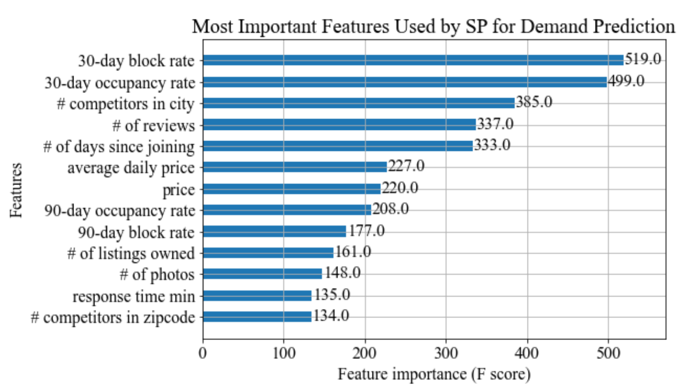


Figure A.4: Most Important features used by the SP algorithm for demand prediction

The second replication step is a "per-listing" regression model to predict the "optimal" price for each listing-night, in which a customized loss function is minimized. The loss function in this step is inspired by the ε-insensitive loss function used in Support Vector Regression (SVR) models. The simple idea behind this step is that listings that are mostly booked at their current prices set by their hosts, might benefit from increasing their prices. However, listings that typically remain unsold, may benefit from decreasing their prices. Formally, the objective of the algorithm in this step is to find a set of parameters θ such that given N training samples $\{x_i, y_i\}_{i=1}^{N}$, where x_i represents input associated with a listing-night and y_i indicates its observed booking status (1 for booked nights and 0 for non-booked nights), the suggested price $f_\theta(x_i)$ minimizes the loss L:

$$L = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{N} [L(P_i, y_i) - f_\theta(x_i)]^+ + [f_\theta(x_i) - U(P_i, y_i)]^+ \tag{A.7}$$

where the superscript + represents the $max\{0,.\}$ operator. The mapping for suggested price, $f_\theta(x_i)$, is defined as $f_\theta(x_i) = P \cdot [1 + \theta_1 \left(q^{\theta_3}\right)^{-q} - \theta_2)$, where P is the representative calendar price set by the host and q is the estimated booking probability at P using the model from the first replication step. In this replication step, the algorithm considers a lower bound, $L(P_i, y_i)$, and an upper bound, $U(P_i, y_i)$, for the optimal price range so that when the suggested price, $f_\theta(x_i)$, lies between the upper and lower bounds, there is no loss. Otherwise, the loss is defined as the distance between the suggested price and the closest bound. The definition of L and U are as follows: $L(P_i, y_i) = y_i . P_i + (1 - y_i). c_1 P_i$ and $U(P_i, y_i) = (1 - y_i). P_i + y_i . c_2 P_i$. For booked listing-nights, $y_i = 1$, the lower bound is the booking price P_i and the upper bound is $c_2 P_i$, where $c_2 > 1$ is a constant. However, for non-booked listing-nights, $y_i = 0$, the lower bound is $c_1 P_i$, where c_1 is a constant within the interval of (0, 1) and the upper bound is the calendar price P_i at which the listing remain unsold. Ideally, the constants c_1 and c_2 can be learned via hyper-parameter search to optimize the outcome for hosts and marketplace. However, because of the limited size of our data, we fix these parameters at $c_1 = 0.95$ and $c_2 = 1.05$.

The definition of representative calendar price, P, is a choice of analyst and different candidates can be used for training, such as the listing's median price over the last 30 days, 60 days, 90 days, or even the whole history of past prices. We experiment with the definition of this parameter and find that the median price over the whole history of past prices works best in the context of our study. The summation in the RHS of equation (A.7) is over a training sample covering a time span of four weeks before to two weeks after the focal night.

To train the model specified in (A.7), we use a constrained non-linear optimization technique, Sequential Quadratic Programming (SQP), and set a bounded range of $(-100,100)$ for θ_1 and θ_2, and a range of (1,2) for θ_3. The trained model is then used to generate (replicated) Smart prices for all listings-nights in the data.

Finally, we compare the replicated Smart prices with the actual Smart prices observed in the data. The mean absolute percentage error (MAPE) for the listings used in the train and test sets of the first replication step is 8.57%, and for the listings kept as hold-out sample is 9.46%. This suggests that our replicated algorithm generates prices that are reasonably close to actual Smart prices of SP during the time of our data.

Section 13. Evidence that Future and Current Smart Prices Are Not Informative

On Airbnb platform, the host can observe current and future Smart prices when they activate SP. A concern here is that even if hosts may have imprecise beliefs about SP prior to its adoption, there is no need for hosts to learn about SP post-adoption, since they can observe the current and future SP prices after adopting it. Thus, the hosts, on each day, may decide based on the observed SP for that day rather than their belief about the SP. Note that this concern rests on the following two assumptions:

a) The smart price on the host's calendar for a given date *s* in the future does not vary over time (i.e., it remains the same across all current dates *t*, where *t* < *s*).

b) The smart price on the host's calendar for the current date t does not vary during the course of that day t.

To see why this concern rests on these assumptions, consider the case in which the smart price for a future date s varies substantially over time. This would imply that the smart price on the host's calendar for a future date s will not be very informative about the price at which the property is ultimately rented on that date s. Similarly, consider the case in which there is significant variation in the smart price for the current date t throughout the course of that day. This implies that the smart price on the host's calendar at a particular time on the current date t is not very informative about the price at which the property is ultimately rented on that date t. Thus, it is unlikely that a host would decide whether to use SP, CP, or BP each day based on the observed SP draw at a particular moment of time during that day. Instead, the host would make this decision each day based on her beliefs about SP's price level on that day.

Based on the above premise, we will address the concern by showing that there is significant variation in the smart price for a future date s over time, and the smart price for the current date t during the course of that day t. Unfortunately, we cannot demonstrate this using our data because we do not observe smart prices for future dates. Additionally, we cannot use the data from other papers that have investigated price frictions on Airbnb (Huang, 2024; Li et al., 2016; Pan and Wang, 2022) because they do not include information on whether a host adopted SP.

Thus, we will provide two additional pieces of evidence suggesting significant variation in the smart price for future date s over time, and the smart price for the current date t over the course of that date t. The first evidence comes from blogs and host forums. The second piece of evidence is based on a real Airbnb listing that we created in August 2025.

First Piece of Evidence: The following references suggest that there is intra-day variation in the smart price:

1. Airbnb Dynamic Pricing: What Does it Say About the Algorithm? https://hometeamluxuryrentals.com/dynamic-pricing: Dynamic pricing ensures that your listing is always priced competitively as market conditions change. Factors such as seasonality, local demand, competitor pricing, and special events are continually analyzed by the algorithm. By adjusting rates daily or *even hourly,* dynamic pricing tools help hosts avoid missed revenue opportunities during peak periods while also preventing vacancies during low-demand times.
2. How often does "smart pricing" adjust rates? https://www.reddit.com/r/AirBnB/comments/xfcccv/how_often_does_smart_pricing_adjust_rates/: "I noticed every time I've wanted to book a stay it ups the price of a place on me within minutes of browsing if I visit a page just a couple of times"
3. Pricing Changes: https://www.reddit.com/r/airbnb_hosts/comments/1jvzitt/pricing_changes/.: "I see the prices going up and down multiple times a day even."

The following references suggest that there is substantial variation in the smart price for a future date s over time:

1. Learning Market Dynamics for Optimal Pricing: https://medium.com/airbnb-engineering/learning-market-dynamics-for-optimal-pricing-97cffbcc53e3 -> "market conditions typically change as booking dates get closer to check-in dates, and, as a result, it is critical for us to account for these changes and help hosts keep their prices optimized for market conditions"
2. https://medium.com/airbnb-engineering/learning-market-dynamics-for-optimal-pricing-97cffbcc53e3: "Guests will continue to make bookings [...] as time progresses and the booking date

gets closer and closer [...]. This booking process reflects the inflow of demand and can be treated as a stochastic arrival process*...* The arrival process is stochastic. For the purposes of optimal pricing, we need to know the distribution of bookings over lead time and not just the average lead time to book... At Airbnb, we have millions of unique listings — each with its own defining characteristics and features that govern the arrival process. This makes every listing a very high-dimensional data point."

<u>Second Piece of Evidence:</u> We created a real Airbnb listing in August 2025 and activated SP for the entire month. This allowed us to track changes in the Smart price of each given future date over time. We checked the host's calendar daily at 12 noon during August 2025. Figure 2 below illustrates the dynamics of future smart price changes in August in the calendar.

Figure 2: How current and future Smart prices change when checking the calendar on three different days

In Figure 2, there are three frames of the host's calendar in August 2025, with the current date varying across them. In these frames, the highlighted box with a black outline is the current date at the time of checking the host calendar. For instance, the current date in the leftmost frame is Aug 1st, in the middle picture it is Aug 5th, and in the rightmost frame it is Aug 15th. The shaded boxes in each frame represent past dates, and the white boxes without a black outline represent future dates. In each box, the price is the smart price for that date. The smart price shown for a past date is the final realized smart price on that date. To understand what we mean by 'the final realized smart price on a given date', consider the case where the date under consideration is Aug 3rd. The smart price for Aug 3rd, once it becomes a past date (as shown in the middle and rightmost frames), is \$97. This is the final realized smart price for Aug 3rd. Why? This is because the smart price fluctuates even within the course of the current day (more on this later), so \$97 represents the last realized smart price by the end of the day on Aug 3 (when Aug 3 was the current date). And this will be the smart price shown on the calendar for Aug 3rd when it becomes a past date (i.e., when the current date on the calendar is Aug 4th or later).

We first examine how the smart price of a future date s varies over time. The three frames in Figure 2 show that the Smart price for a future date on the host's calendar changes substantially over time. For instance, consider the smart price for a future date, August 16th, across the three frames: it is \$101 in the leftmost frame in which the current date is Aug 1st, \$90 in the middle frame in which the current date is Aug 5th, and \$109 in the rightmost frame in which the current date is Aug 15th. This suggests the host cannot treat the smart price of a future date as fixed.

To analyze this more rigorously, we next quantify the extent of variation in the smart price of a given future date *s* over time. To do so, we recorded all future Smart prices (across all current dates) for our Airbnb listing over the period August 1-30, 2025, and calculated the variance of the Smart price over time for each future date *s*. The average variance across all future dates *s* is 295.5. This suggests significant variation in the Smart price for a given future date over time.[1]

We next examine whether the smart price of a current date *t* varies over the course of that day. Observe in Figure 2 that the smart price shown for a current date *t* (which, as discussed earlier, was recorded at 12 noon on that date *t*) can be different from the final realized smart price for that same date *t*. For instance, consider August 1st. When Aug 1st is the current date, the smart price for Aug 1st recorded at 12 noon on that day is $95 (as shown in the leftmost frame). However, when August 1st becomes a past date, the smart price shown in the calendar for Aug 1st is $84 (as shown in the middle or rightmost frame). As discussed earlier, this $84 price represents the final realized smart price at the end of the day on Aug 1st. This suggests there is significant variation in the smart price of a current date *s* over the course of that day. To analyze this more rigorously, we calculated the variance of the difference in 'the smart price at 12 noon for a current date *t* on the calendar' and 'final realized price on that date *t*'. The variance over all days in Aug is 195.3. This suggests that the smart price on the host's calendar at a particular time on the current date may not be very informative about the price at which the property is ultimately rented on that date.

The two pieces of evidence suggest significant variation in the smart price of a future date over time and in the smart price of a current date over the course of that date. Thus, in our paper, we abstract away from the current and future smart prices shown on the host's calendar and instead focus on the host's day-to-day learning about SP from the final 'recorded' smart prices. As discussed in the paper, the final recorded smart price for date *t* is either the smart price at which the property was rented on that date, or the final realized smart price on that date if the property was not rented on that date.

Section 14. Reduced-Form Evidence Comparing BP and CP before and after SP

In this section, we compare the price levels of BP and CP set by the hosts in the pre-SP period with the respective price levels of BP and CP set by the hosts in the post-SP period. We run the following regression:

$$p_{jt} = Const + \alpha\, CP_{jt} + \beta\, CP_{jt} \times PostSP_t + \gamma\, BP_{jt} \times PostSP_t + Listing_j + Month_t + Weekend_t + \varepsilon_{jt}$$

where BP_{jt} and CP_{jt} are binary variables that represent whether listing j at period t uses BP and CP, and $PostSP_t$ is a binary variable that represents whether period t is after the introduction of SP. $Listing_j$ is a listing fixed effect, and $Month_t$ and $Weekend_t$ are time fixed effects. The estimated coefficients are presented in Table A.10. This implies that the difference between the level of BP before and after SP is not statistically significant. The same conclusion holds for the level of CP before and after SP.

---

[1] We also investigated whether the smart price for a future date varies systematically with respect to the number of days between the current and the future date, which may stem from dynamic pricing (whereby SP changes the price for a future booking date as the time difference between the current and the future booking date changes). To do so, we ran a regression in which we regressed the difference between the smart price for a future date *s* and the final realized price on that date *s* on the number of days between the future date *s* and the current date *t* (i.e., *s* - *t*). The unexplained variation in this regression is 98.1%, implying that most of the variation in the difference in the smart price for a future date and its final realized price stems from reasons unobserved to the host, such as updates to the demand forecast for that date, changes in neighborhood search volume for that date, etc.

Table A.10: Regression coefficients for comparing CP and BP before and after SP

	(1)
VARIABLES	Blocked
CP	3.03**
	(1.31)
CP × PostSP	0.19
	(1.50)
BP × PostSP	1.36
	(0.89)
Constant	202.35***
	(0.60)
Observations	2,784,235
R-squared	0.906
Listing FE	YES
Month FE	YES
Weekend FE	YES

Clustered robust standard errors in parentheses *** $p<0.01$, ** $p<0.05$, * $p<0.1$

We re-run the above regression using data only from adopters who stop using SP after trying it out and report the results in Table A.11 below. The conclusion remains similar.

Table A.11: Regression coefficients for comparing CP and BP before and after SP (for adopter who stop using)

	(1)
VARIABLES	Blocked
CP	0.38
	(3.00)
CP × PostSP	- 2.66
	(2.90)
BP × PostSP	- 1.42
	(2.98)
Constant	172.29***
	(1.84)
Observations	294,036
R-squared	0.849
Listing FE	YES
Month FE	YES
Weekend FE	YES

Clustered robust standard errors in parentheses *** $p<0.01$, ** $p<0.05$, * $p<0.1$

Section 15. How the Three Broad Factors for Not Adopting SP follow from the Main Tradeoffs a Host makes when deciding to adopt SP.

As discussed in the paper, we consider a set of 'broad' factors/reasons for low SP adoption, namely: biased prior beliefs about SP's performance, hosts' adoption costs, and SP's sub-optimality for hosts. These three broad factors follow directly from the main tradeoffs of a host's adoption decisions in practice. In what follows, we will explain why these three broad sets of factors for not adopting SP are collectively exhaustive, and how they follow directly from the main trade-offs faced by the hosts.

We begin with the main trade-offs we consider in our paper when a host decides whether to adopt SP in period *t*. When a host decides whether to adopt SP in period *t*, the trade-offs she faces are: (1) the adoption cost of activating SP vs. (2) the host's beliefs about the difference in payoffs (without considering the adoption cost) she would get if the price were set by SP in period *t* as opposed to if she were to set it herself using CP/BP.

Note that point (2) is simply the host's belief about the relative performance of SP as compared to CP/BP, which can be further split into two sub-parts: (a) the true relative performance of SP compared to CP/BP, and (b) the difference between the 'hosts' beliefs about the performance of SP' and 'the true performance of SP'. Note that (b) is simply the extent of bias in the host's prior beliefs about the performance of SP. Next, recall from section 3.2 of the paper that CP yields higher profits than SP. Thus, (a) would be synonymous with the extent of sub-optimality of SP.

Collecting all these tradeoffs together, we see that the host's decision to adopt SP in period t will depend on the following three categories of 'broad' factors:

(1) Adoption cost of activating SP
(2) The extent of sub-optimality of SP
(3) The extent of bias in the host's prior beliefs about the performance of SP.

The above discussion summarizes all the main trade-offs a host faces when deciding whether to adopt SP. Therefore, factors (1) – (3) constitute a collectively exhaustive set of 'broad' factors that impact the host's decision to adopt SP. Why? That is because these 3 broad categories of factors follow directly from the main trade-offs the hosts face. This is why in our paper, we categorize the potential reasons for the low adoption rate of SP into these three broad categories (1)–(3): adoption costs, biased prior beliefs about SP's performance, and SP's suboptimality

Section 16. Counterfactual Analyses with Minimum Bounds on SP

At Airbnb, a host can set a minimum and maximum price for SP. Unfortunately, we do not observe these bounds in the data. This is unlikely to affect our estimation results, since the bounds on SP are automatically incorporated into the smart price draws observed in the data. However, this may not hold for counterfactual analyses, as the results can change when we impose minimum and maximum bounds on smart prices for each host.

Comparing the impact of imposing minimum versus maximum bounds on SP in counterfactual analyses, we would expect the results to be more sensitive to the minimum bounds than to the maximum bounds. This is because we expect the hosts' maximum SP bound to fall within the range of their Custom and Base prices. The reason for this follows from our reduced-form analysis in Section 3.2, which shows that the hosts' Base and Custom prices are higher than the Smart prices. This suggests that the maximum bound of the SP set by hosts will most likely not be binding as far as the draws of SP are concerned in the counterfactuals. Next, consider the minimum bound for SP when conducting the counterfactuals. This could be a concern, since SP's prices are significantly lower than the BP or CP prices set by hosts. To address this point, we re-ran all of our counterfactual analyses by setting a minimum SP price for each host. For each property, we use the lowest price ever set for that property throughout the observed history as the minimum price for SP.[2] We present our new counterfactual results in Table A.12. The results are qualitatively similar to our original counterfactuals, suggesting they are fairly robust to excluding the maximum and minimum bounds for SP. This suggests that omitting the minimum and maximum bounds will not significantly affect the main results of our paper.

Table A.12. Summary of counterfactuals with min SP bound equal to min historical price

Scenario	SP adoption rate	Average daily SP usage rate	Change in average annual host profit	Change in annual platform revenue
Current Situation (status quo)	20.7%	7%	-	-

[2] The reason why we have taken the lowest price charged by a host as the minimum bound for SP for that host is because the Maximum Likelihood Estimate of a minimum bound (or a parameter defining the lower limit of a distribution's support) is generally the sample minimum, $X_{(1)} = \min(X_1, \dots, X_n)$

No adoption cost	39%	11%	+4%	+3%
Educate hosts to correct their beliefs about SP	60%	16%	+13%	+8%
Re-trained SP using marginal costs + Pessimistic prior beliefs	28%	7%	+16%	+4%
Re-trained SP using marginal costs + Correct hosts' beliefs	96%	48%	+94%	+30%

Section 17. Text Analysis of Discussions about SP on Three Major Online Forums

We conducted a large-scale text analysis of discussions about Airbnb Smart Pricing (SP) on major host-focused online forums. The goal of this analysis is to systematically identify the concerns that hosts associate with SP that may contribute to pessimistic prior beliefs. This analysis is conducted in 5 steps. We start with a large corpus of host discussions from three major online forums (AirHostsForum, Reddit, and Airbnb Community Center). Over the steps 1-5, which we discuss below, we progressively narrow the entire data to the specific content directly related to Airbnb's SP tool, extract the relevant discussion context, identify discussions expressing concerns about SP, and classify them into specific categories of concerns expressed by hosts about SP. These specific categories of concerns will then help us understand the underlying reasons for their pessimistic prior beliefs about SP.

Step 1. We collected 435,165 unique posts from three major host-focused online forums: AirHostsForum (269,767 posts), host-oriented Reddit communities (164,559 posts), and the Airbnb Community Center (835 posts). We tried to scrape as many posts as possible on these forums to start with. We then did a document-level screening to identify posts that directly discuss SP. Specifically, from the raw archive, we retained posts whose combined title and body contained at least 30 characters and matched an expanded dictionary of Smart Pricing and pricing-algorithm language (direct phrases such as "smart pricing," "pricing algorithm," "automated pricing," "Airbnb algorithm," plus comparative tool names). This process yielded 10,533 posts.

Step 2. Next, each retained post was split into candidate sentences (period, question mark, or exclamation followed by whitespace and a capital letter, or line breaks; fragments under 16 characters discarded). A sentence was kept only if it passed either of these criteria: (a) one of 20 exact substrings tied to Airbnb's tool (see Table A.14), or (b) one of eleven contextual regular expressions requiring tight co-occurrence of Airbnb, pricing verbs, algorithm language, incentive language, or third-party tools with explicit SP wording in the same sentence (Table A.15). Posts whose every retained SP-tagged sentence names only a third-party pricing tool (PriceLabs, Wheelhouse, Beyond Pricing, Hospitable, Hostaway, Guesty, Lodgify, OwnerRez, Smartbnb, Rabbu, AirDNA, RentalsUnited, MyVR, iGMS, and Hostfully) without any Airbnb SP language in that same sentence are also dropped in this same pass. The sentence-relevance filter and competitor-only exclusion are applied jointly in a single step, yielding 3,751 posts with at least one qualifying SP sentence. We then restrict to the three host-focused forums and deduplicate by post identifier, URL, and normalized text, removing 315 posts. The analytic sample is n = 3,436 posts (AirHostsForum 1,638; Reddit 1,617; Airbnb Community 181).

Step 3. For each SP-relevant sentence in a post, we form a context span consisting of that focal sentence plus up to two preceding and two following sentences drawn from the post body only (capped at 1200 characters). Thread titles are excluded because in multi-comment threads the title often reflects the original poster while follow-up comments address different sub-issues. This produces 5,116 context spans from

3,436 posts.

Step 4. SP sentences are embedded with the publicly released SentenceTransformer model "all-MiniLM-L6-v2" and classified against hand-written negative, positive, and neutral prototype anchors (see Table A.16). A sentence is labeled negative if its cosine similarity to the negative prototype strictly exceeds that to the positive prototype and exceeds the neutral prototype by more than 0.04; the symmetric rule applies for positive; otherwise the sentence is labeled neutral. A post is labelled "mostly_negative" if at least 55% of its SP sentences are negative, "mostly_positive" if at least 55% are positive, and "mixed" otherwise. In our sample, 46.1% of posts are mostly negative toward SP and 45.8% have mixed sentiment, confirming that the majority of forum discussion is skeptical toward SP. The 8.1% of posts classified as mostly positive (n = 278) are excluded from reason classification in Step 5, as their context spans are unlikely to express pessimistic mechanisms.

Step 5. Each context span is classified into seven predefined categories of concern regarding SP (see Table A-17). We embed each span with the same all-MiniLM-L6-v2 model and compare it to L2-normalized centroid embeddings built from hand-written anchor sentences for each category (see Table A-18). A span is assigned to a category if its cosine similarity to that category's centroid is at least 0.35. A span may receive more than one label when it is a genuine near-tie between two categories (second similarity within 0.035 of the top). Spans from the 278 posts classified as mostly positive in Step 4 are excluded before classification. Of the remaining 4,775 spans, those for which no category exceeds the threshold are excluded from mention counts (210 of the 4,775 spans submitted to classification, 4.4%). A total of 4,565 spans received at least one reason label (average 1.4 reasons per labeled span).

Results: Let M_g be the number of context spans in which reason g is expressed. For each category of concern regarding SP, we report M_g and its share of total mentions, as shown in Table A.13 below. Because labeling can be multi-label, the sum of M_g over categories can exceed the number of spans. Total mentions of concerns towards SP are 6,390. Representative excerpts from each category are provided in Table A-19.

Table A.13. Concerns expressed in host discussions about Smart Pricing Tool (mention-level counts)

Concerns category	Mentions	% all mentions
Perceived Incentive misalignment (Airbnb vs. host)	2,130	33.3
Low displayed values of SP prices	1,273	19.9
Bad outcomes for hosts after using SP	1,023	16
Hosts preferring third-party tools	833	13
Confusion: SP vs. price tips	486	7.6
Algorithm opacity	324	5.1
Adoption cost/hassle	321	5

The above analysis shows that the main concern about SP is to do with the perceived misalignment of incentives between hosts and the platform. That is, hosts believe that SP serves the platform's objective of maximizing platform revenue rather than the host's objective of maximizing the host's profits. This reason falls under the category of pessimistic/biased prior beliefs about SP. To see why that is so, recall that if a stated reason given by hosts for not adopting SP is related to hosts' beliefs about a shortcoming of SP, and if those beliefs about the shortcoming of SP are incorrect, then that reason would fall under the broad category of 'biased beliefs about SP's performance'. Recall that, as per the reduced form evidence presented in section 3.2 of the paper, SP does not maximize the platform's revenue. This suggests that the hosts' belief that 'SP maximizes platform revenue' is incorrect, which in turn suggests that the host's beliefs about misalignment of incentives would fall under biased prior beliefs of hosts about SP's performance.

Other than perceived incentive misalignment, the opacity of SP also contributes to biased beliefs about it. This is because not understanding the logic behind SP can lead to skepticism about its performance, which then leads to biased beliefs about it.

The other concerns are not related to pessimistic prior beliefs. The reasons are as follows.

- The concern "hosts preferring third-party tools" is not relevant to the span of our data, since most third-party pricing tools on Airbnb were introduced after 2018.
- The concern, "Bad experience after activating SP," does not lead to pessimistic prior beliefs because it pertains to the post-adoption period, whereas prior beliefs are formed before trying SP.
- Consider the concern, "Confusing Smart Pricing with Price Tips." Price Tips were introduced by Airbnb before Smart Pricing. Price Tips are suggested prices that appear whenever a host manually enters a custom price. These are based on factors such as location, amenities, past bookings, and prices charged in comparable listings. The key differences between Price Tips and Smart Pricing are: (i) While Smart Pricing is automated, Price Tips are not. Instead, Price tips are more like a nudge: "A competitive rate for this date might be *X*." (ii) Smart Pricing tries to continuously optimize bookability by adjusting the price based on real-time changes in demand, such as the number of people currently searching the property's webpage. On the other hand, Price Tips focuses more on market trends and past bookings than minute-by-minute search volumes. (iii) As per the anecdotal evidence on Airbnb host forums[3], Price tips are generally 20% to 30% higher than the Smart Price levels. Point (iii) implies that if hosts confuse Smart Pricing with Price Tips, they would incorrectly infer that the Smart pricing levels are much higher than they truly are. This would suggest that Price tips would not lead to pessimistic prior beliefs about the smart price levels.
- Consider the concern, "Low displayed values of SP prices." Note that the low displayed value of the SP price (compared to what the host typically charges) could either be the SP price level after adopting SP (in which case, it would play no role in the formation of prior beliefs) or before adopting SP, where the host would have switched on SP for the first time, observed all the current and future SP prices, and then formed prior beliefs about SP's price levels before deciding whether to use SP for at least one night. As we show in the analysis below, the displayed SP price for a current or future date, on average, is significantly greater than the final realized SP price on that date. This would suggest that the low current and future values of SP prices would not lead to pessimistic prior beliefs about the (finally realized) SP's price levels (they would instead lead to optimistic prior beliefs).

To provide evidence for the claim discussed above, we use the data collected in section 13 of the web appendix and run the following regression to examine whether the displayed SP price for a current or future date is, on average, significantly higher than the final realized SP price on that date. For each calendar date t, we calculate the difference between the Smart price shown on the calendar for date t (where t could be a date either in the future or could be the current period) collected during the period before calendar date t is reached) and the final realized Smart price for that date t, and regress this variable on the lead time, which is equal to the time difference between date t and the current date s (where date s is the current date on which the future SP price for date t is recorded. The regression shows that the estimate of the intercept is 14.71 (se = 1.61) and the estimate for the coefficient for lead time is 0.37 (se = 0.13). Note that both are positive and statistically significant ($p < 0.01$). This implies that the displayed future SP price

[3] Solved: Price tips - Airbnb Community; Do you use Smart Pricing as a host? : r/AirBnB

of a future date *t* is, on average, higher than the final realized SP price on that date *t*. This suggests that observing future SP prices is unlikely to lead to pessimistic prior beliefs about SP's price level.

Table A.14. Exact substrings — Pass 1 of SP relevance gate

#	Substring
1	"smart pricing"
2	"smartpricing"
3	"smart price"
4	"airbnb's pricing tool"
5	"airbnb pricing tool"
6	"airbnb's algorithm"
7	"airbnb algorithm"
8	"airbnb's pricing algorithm"
9	"airbnb's dynamic pricing"
10	"pricing algorithm"
11	"automated pricing"
12	"algorithmic pricing"
13	"ai pricing"
14	"machine learning pricing"
15	"algorithm set"
16	"algorithm suggest"
17	"algorithm lower"
18	"algorithm raise"
19	"algorithm price"
20	"price tips"

Table A.15. Contextual regular expressions — Pass 2 of SP relevance gate

<table>
<tr><th>#</th><th>Pattern</th></tr>
<tr><td>1</td><td>\bairbnb.{0,25}(sets?|lower|raise|adjust|change|suggest|recommend|calculat|automat|tool|ai\b)</td></tr>
<tr><td>2</td><td>airbnb.{0,40}dynamic pricing|dynamic pricing.{0,40}airbnb</td></tr>
<tr><td>3</td><td>(algorithm|automated pricing|automat\w+|ai tool|machine learning).{0,25}(price|rate|nightly|fee)</td></tr>
<tr><td>4</td><td>(price|rate|nightly|fee).{0,25}(algorithm|automated|automat\w+|suggested by airbnb|recommended by airbnb)</td></tr>
<tr><td>5</td><td>\bsp\b.{0,20}(price|rate|low|high|off|on|enable|disable|tool|feature)</td></tr>
<tr><td>6</td><td>(turn(ed)? off|disable[d]?|switch(ed)? off).{0,30}(smart pricing|airbnb.{0,15}pricing|pricing algorithm|automated pricing)</td></tr>
<tr><td>7</td><td>(suggested|recommended)\s+price.{0,30}(airbnb|algorithm|smart|automated)</td></tr>
<tr><td>8</td><td>(airbnb|algorithm|smart pricing).{0,30}(suggest |recommend).{0,20}(price|rate|nightly)</td></tr>
<tr><td>9</td><td>airbnb.{0,20}(want|care|incentive|interest|benefit|profit).{0,25}(book|host|revenue|money)</td></tr>
<tr><td>10</td><td>(pricelabs|wheelhouse|beyond pricing|rabbu)\b.{0,60}(smart pricing|airbnb.{0,15}pric|algorithm|sp\b)</td></tr>
<tr><td>11</td><td>(smart pricing|airbnb.{0,15}pric|algorithm|sp\b).{0,60}(pricelabs|wheelhouse|beyond pricing|rabbu)</td></tr>
</table>

Table A.16. Sentiment prototype anchor sentences

Prototype	#	Anchor sentence
Negative	1	Smart pricing sets my rates way too low and I lose money.
Negative	2	I turned off smart pricing because it was hurting my revenue.
Negative	3	Airbnb's pricing algorithm undervalues my listing.
Negative	4	Smart pricing is terrible and I don't trust it.
Negative	5	The algorithm doesn't know my local market at all.
Negative	6	Smart pricing suggested a price that was insultingly low.
Negative	7	I disabled smart pricing after it cost me bookings.

Negative	8	Airbnb only cares about bookings, not my income.
Negative	9	Smart pricing is a black box — I have no idea how it works.
Negative	10	The algorithm kept lowering my price without my permission.
Negative	11	I switched to PriceLabs because smart pricing was awful.
Negative	12	Smart pricing does not account for local events or seasons.
Negative	13	The suggested prices are completely unrealistic and too low.
Negative	14	I hate smart pricing — it makes me earn less per night.
Negative	15	Airbnb's incentives are misaligned with mine as a host.
Positive	1	Smart pricing works great for me and I get more bookings.
Positive	2	I love using smart pricing — it saved me time.
Positive	3	The pricing algorithm helped me fill my calendar.
Positive	4	Smart pricing is accurate and reflects market demand.
Positive	5	I trust the algorithm to set the right price.
Positive	6	Smart pricing increased my occupancy rate significantly.
Positive	7	The suggested prices are reasonable and competitive.
Positive	8	I recommend smart pricing to new hosts.
Positive	9	Smart pricing helped me earn more overall.
Positive	10	The algorithm does a good job adjusting for demand.
Neutral	1	I use smart pricing but I also set a minimum price.
Neutral	2	Smart pricing is available in the settings.
Neutral	3	You can turn smart pricing on or off in your dashboard.
Neutral	4	Some hosts use smart pricing, others prefer manual pricing.
Neutral	5	Smart pricing adjusts prices based on demand.

Table A.17. Reason category definitions

Label	Definition / coding note
Perceived Incentive misalignment (Airbnb vs. host)	Platform optimizes bookings, fees, or platform revenue—not host profit. Includes complaints about host service fees or commissions when framed as misaligned incentives.
Low displayed/suggested SP prices	The SP price the host sees—displayed on the calendar, the suggested/recommended rate—is too low or below market. Distinct from realized bad outcomes after running SP; excludes minimum-floor violations and displayed-vs-set sync bugs.
Bad outcomes after using SP	Lower revenue, occupancy, or bookings after trying SP. Includes cases where SP goes below the host's minimum.
Confusion: SP vs. price tips	Conflates Smart Pricing with separate Airbnb price-tip nudges.
Adoption cost / hassle	Friction or risk before first enabling SP (setup mistakes, fear of turning on); not ongoing uncertainty about how the algorithm works.
Prefer third-party tools	Favors PriceLabs, Wheelhouse, or similar over Airbnb SP.
Algorithm opacity	SP behaves like a black box—hosts cannot tell when, why, or how well it adjusts; includes uncertainty about whether SP 'knows' local demand, events, or seasonality; trial-and-error toggling to see when it works.

Table A.18. Reason category anchor sentences

Category	Anchor sentence
Perceived Incentive misalignment (Airbnb vs. host)	Airbnb just wants more bookings, they don't care about my revenue.
	Smart pricing is designed to benefit Airbnb, not hosts.
	The algorithm prioritizes occupancy over my profit per night.
	Airbnb's incentives are misaligned with mine as a host.
	They push prices toward headline occupancy, not toward what is best for my revenue.
	Airbnb profits from volume so they want lower prices to get more bookings.
	Airbnb increased host fees and smart pricing does not compensate.
	Guest service fees and host commission make the listed smart price meaningless.

	UK hosts are getting hit with higher Airbnb commission on bookings.
Low displayed/suggested SP prices	When I look at my calendar, smart pricing is showing prices far below what my listing is worth.
	The smart price suggested for future dates is well below market.
	Smart pricing's recommended nightly rate is insultingly low.
	The price displayed for upcoming bookings under smart pricing is too cheap.
	Smart pricing keeps suggesting rates lower than comparable listings in my area.
	On my calendar, smart pricing sits at a price far lower than I would ever set.
	I set a minimum price but smart pricing still shows the minimum for every day.
	Why does smart pricing show my floor price on every available day?
	My calendar is stuck at the minimum even with smart pricing on.
Bad outcomes after using SP	My revenue dropped after I enabled smart pricing.
	I earned less money with smart pricing than with manual pricing.
	Smart pricing hurt my occupancy and I turned it off.
	I lost money using Airbnb's automated pricing.
	I had a terrible experience with smart pricing and turned it off immediately.
	Smart pricing ignores my minimum nightly rate.
	The algorithm keeps pricing at my minimum and never goes higher.
	The price shown before booking was higher than what I actually got paid.
	Displayed smart pricing was misleading compared to the final payout.
	Guests saw one price but the realized rate after fees was much lower.
	Airbnb shows attractive future prices but the actual smart price realized lower.
	What you see in the calendar preview is not what you end up earning.
Confusion: SP vs. price tips	Are price tips the same as smart pricing? The interface confuses me.
	Airbnb's price tips seem wrong and I don't know how they relate to smart pricing.
	Smart pricing and those little price suggestions feel like two different systems.
	The price tip numbers disagree with what smart pricing is showing.
	I hate the suggested prices by Airbnb along with their harassing emails.
Adoption cost / hassle	I have not turned on smart pricing because I don't have time to monitor it.
	Turning on smart pricing feels risky; I don't want to fix bad prices.
	The setup is too much hassle for something that might hurt my revenue.
	I delay enabling smart pricing because onboarding is confusing in the dashboard.
Prefer third-party tools	I switched to PriceLabs and it is much better than Airbnb's smart pricing.
	Wheelhouse gives me way more control than smart pricing.
	I use PriceLabs instead of smart pricing because it customizes to my market.
	The fee I pay to a third-party pricing tool is worth it compared to Airbnb smart pricing.
Algorithm opacity	Smart pricing is a black box with no transparency.
	I don't trust Airbnb's algorithm to know my local market.
	The suggested prices seem random and disconnected from reality.
	There is no way to see why the algorithm chose this price.
	Smart pricing ignores local events and seasonality in my area.
	Is smart pricing really all that smart? I don't think so.

Table A.19. Representative context-span excerpts. Each row shows the focal SP sentence plus up to two neighboring sentences from the post body (titles excluded).

Reason category	Ex.	Context span excerpt
Perceived Incentive misalignment (Airbnb vs. host)	1	Airbnb smart pricing is designed to make THEM money and keep your rooms occupied 100%. This is not usually what is best for you. It's usually waaaay undervalued.

	2	Smart pricing sucks. Airbnb should feel ashamed for how bad it is when they have so much data to build good models. Friendly reminder that Smart Pricing is designed to make *Airbnb* more money, not make *you* more money. Yes, use the 3rd party pricing services.
Low displayed/suggested SP prices	1	I just saw for July that it smart priced at $60 and I'll easily get over 200 for that night if I price it myself once I take smart pricing off.
	2	No matter what you set it at, the app will suggest lower. If you set it for $40 a night, it might suggest $30, and if you raise it to $60 right after, it might suggest $40! You can look ahead on the calendar and see what Smart Pricing is setting.
Bad outcomes after using SP	1	I get the request all the time to add smart pricing. I did mid year and lost $3.000- because smart pricing overrides your seasonal school holiday prices etc. We should be able to charge what we think our homes are worth and what we offer.
	2	I got fully booked for August in 3 days when I naively used Smart Pricing as a first time host in June. I lost 4k compared to what I should have been charging and I got horrible guests who nitpicked, damaged my stuff and stole towels and my kitchen gadgets.
Confusion: SP vs. price tips	1	Airbnb's price tips are absolutely worthless and I've kept every option under the notification marketing tab off for years, I prefer to not even see them. I think many people confuse price tips with smart pricing - completely different options. I personally have used smart pricing since they first made it available and would be lost without it. If you don't set it up properly when you turn it on, of course it will bo
	2	DozerPug: think many people confuse price tips with smart pricing - completely different options. Yes, that $19 is their "price tip". I've never been interested in using smart pricing- my place is the same price every day of the year.
Prefer third-party tools	1	Couldn't agree more on Beyond Pricing. It is soooooo much better than smart pricing. We saw a 35-40% increase in revenue which easily paid for their 1% fee.
	2	I use beyondpricing.com It's much better than smart pricing as it takes into account events in town.
Algorithm opacity	1	Hi everyone, I am quite shocked to find out that the base price defined in my listing does not apply to the days in the calendar. I found about this yesterday while experimenting to use the `Smart price` feature. I pressed it and airbnb changed all the prices of my listing according to their algorithm. I was just curious of what were the prices and I wanted to try.
	2	Yup, all it takes is a quick search of the people around you, looking at who has similar listings, and seeing that a VERY large percentage of them are priced WELL above what smart pricing recommended for your nightly rate. Smart pricing does not even take local events into consideration. It's not that hard to see smart pricing for what it is, a way for ABB to prioritize high occupancy rates over host revenue so they

References:

- Austin PC (2013). The performance of different propensity score methods for estimating marginal hazard ratios. Stat Med. 32(16):2837–2849. doi:10.1002/sim.5705
- Austin, P.C., E.A. Stuart (2015), Moving towards best practice when using inverse probability of treatment weighting (IPTW) using the propensity score to estimate causal treatment effects in observational studies.

Statistics in Medicine 34, 3661–3679 (2015).
- Autor, D.H. (2003), "Outsourcing at Will: The Contribution of Unjust Dismissal Doctrine to the Growth of Employment Outsourcing," Journal of Labor Economics 21(1):1-42
- Borusyak, K., Jaravel, X., & Spiess, J. (2021). Revisiting event study designs: Robust and efficient estimation. working paper
- de Chaisemartin, C., & D'Haultfoeuille, X. (2020). Two-way fixed effects estimators with heterogeneous treatment effects. American Economic Review , 110(9), 2964–2996.
- de Chaisemartin, C, D'Haultfoeuille, X (2024). Difference-in-Differences Estimators of Intertemporal Treatment Effects, Review of Economics and Statistics, forthcoming
- Fradkin, Andrey (2017), Search, Matching, and the Role of Digital Marketplace Design in Enabling Trade: Evidence from Airbnb (March 21, 2017). https://ssrn.com/abstract=2939084
- Goodman-Bacon, A. (2020). Difference-in-differences with variation in treatment timing. working paper .
- Guo, S. & Fraser, M. W. (2015). Propensity score analysis: Statistical methods and applications (2nd ed.). Thousand Oaks, CA: Sage Publications, Inc.
- Imai, S., Jain, N., & Ching, A. (2009). Bayesian estimation of dynamic discrete choice models. Econometrica, 77, 1865-99.
- Masten, Matthew, and Alexandre Poirier (2018). "Identification of Treatment Effects Under Conditional Partial Independence." Econometrica 86 (1): 317–51.
- Rosenbaum, P.R. (1987), Observational Studies. 2nd ed. (New York: Springer).
- Rosenbaum, P.R., D.B. Rubin (1983), The central role of the propensity score in observational studies for causal effects. Biometrika 70, 41–55.
- Roth, J., Sant'Anna, P., Bilinski, A., Poe, J. (2023), What's Trending in Difference-in-Differences? A Synthesis of the Recent Econometrics Literature. Journal of Econometrics, vol. 235 (2), p. 2218-2244, 2023
- Sun, L., & Abraham, S. (2021). Estimating dynamic treatment effects in event studies with heterogeneous treatment effects. Journal of Econometrics, forthcoming .
- Train, K. (2009). Discrete choice methods with simulation. Cambridge university press
- Ye, P., Qian, J., Chen, J., hung Wu, C., Zhou, Y., Mars, S. D., . . . Zhang, L. (2018). Customized regression model for Airbnb dynamic pricing. The 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '18), ACM, New York, NY, USA, 932-940
- Zalmanson, Lior and Proserpio, Davide and Nitzan, Irit (2018), Cancellation Policy as a Signal of Trust and Quality in the Sharing Economy: The Case of Airbnb
- Zhang, S., Mehta, N., Singh, P. V., & Srinivasan, K. (2021). Frontiers: Can an ai algorithm mitigate racial economic inequality? an analysis in the context of airbnb. Marketing Science, 40 (5), 813–820