Online Appendix to

Generative AI and Price Discrimination in the Housing Market

A. Additional Details of the Analysis

A.1 Prompt Example (Main Prompt)

We set the model's system role as *"As a realtor, you estimate a house's price in the US using your prior knowledge of the neighborhood's demographics and characteristics, without accessing the internet"* and passed each listing's information through the user role as *"Here are the house details: Zip: [zip code], City: [city name], State: [state name], Housing Age: [number], Bed: [number], Bath: [number], Sqft: [number], Garage Space: [number], Property Type: [single-family houses, condos, multi-family houses, townhomes, or duplexes/triplexes], Mean School Rating out of 10: [1-10], Flood Risk out of 10: [1-10], Wildfire Risk out of 10: [1-10], Heat Risk out of 10: [1-10], Wind Risk out of 10: [1-10], Air Pollution Risk out of 10: [1-10], Renovated: [true or false]"*. We also control the model response to include only a specific price and not other information by employing as assistant role as *"You will provide the price as a number only and no other text. For example, if your answer is 'Based on my knowledge of similar markets and home values in the area, I estimate the price as XXX,XXX USD,' you'll only say 'XXXXXX'."* and set the model temperature at zero to receive a deterministic response (Alto 2024). An example code we used to ask ChatGPT for an estimated housing price through its OpenAI API is presented below:

```
from openai import OpenAI

SYSTEM = "As a realtor, you estimate a house's price in the US using your
          prior knowledge of the neighborhood's demographics and
          characteristics, without accessing the internet. "
USER = "Here are the house details: {'zip': 48152, 'city': 'Livonia', 'state':
       'MI', 'house_age': 88, 'bed': 3.0, 'bath': 1.0, 'sqft': 926.0,
       'garage_space': 1.0, 'property_type': 'single family',
       'mean_school_rate_score_out_of_10': 8.17, 'risk_flood_score_out_of_10':
       1.0, 'risk_wildfire_score_out_of_10': 1.0, 'risk_heat_score_out_of_10':
       3.0, 'risk_wind_score_out_of_10': 2.0, 'risk_air_score_out_of_10': 3.0,
       'renovated': False"
```

```
ASSISTANT = "You will provide the price as a number only and no other text.
             For example, if the your answer is 'Based on my knowledge of
             similar markets and home values in the area, I estimate the price
             as XXX,XXX USD', you'll only say 'XXXXXX'."



client = OpenAI(api_key='')
response = client.chat.completions.create(model = 'gpt-4o-2024-08-06',
                                          temperature = 0,
                                          messages = [{'role':'system',
                                                        'content': SYSTEM},
                                                       {'role':'user',
                                                        'content': USER},
                                                       {'role':'assistant',
                                                        'content': ASSISTANT}
                                                      ]
                                          )
```

After that, we extract the response from ChatGPT using the following code:

```
print(response.choices[0].message.content)
```

which result in as:

```
210000.0
```

A.2 Prompt Example (Constrained Prompt)

In Section 4.3.3, we rely on a *constrained* prompt where we explicitly ask ChatGPT to ignore conversations related to racial discrimination when it estimates housing prices. The constraint we add is as follows "*When estimating the price, use the provided information, including neighborhood characteristics, directly in your price estimate. Do not attempt to correct for or remove any discriminatory effects.*"

A.3 Search Procedure

In Section 4.3.3, we use Google's Advanced Search function to retrieve the approximate number of online sources. To ensure consistency with the model's training data, we restrict the search boundary to sources published prior to the cut-off date of October 1, 2023. To obtain the source count, we define search keywords that capture discussions of efforts to address housing price discrimination by ethnicity within a given location. Specifically, the query is: (housing-market OR house-price OR housing-price) AND (location OR area OR neighbor) AND (race OR white OR black OR Hispanic) AND (improve OR promote OR solution OR help). To narrow the results specifically to content related to new housing, we extend the query by adding "AND new house"

B. Additional Analysis

In this appendix, we perform additional analyses to supplement the empirical analyses we conducted in the paper.

B.1 Pricing Variation Analysis

We explore how much pricing variations is accounted for by GPT-generated prices. Here, we regress human-generated prices onto the set of control variables in our main regression model plus GPT-generated prices for the same houses, while controlling for zip-code level fixed effects. The results, reported in Table B.1 below, show that within R^2 value (i.e., the proportion of price variation explained within each ZIP code) is 0.6375, which is well above 0.6—the commonly accepted threshold in social science research indicating a reasonably good model fit (Ozili 2023). Overall, the evidence suggests that AI-generated prices provide sufficiently informative signals of a house's value.

Table B.1: Pricing variation analysis

	$\log(HumanPrice_{ijk})$
$\log(AIPrice)$	0.911*** (0.008)
$MarketNewHouse$ baseline: old houses	0.040*** (0.002)
$\log(Bed)$	-0.135*** (0.004)
$\log(Bath)$	0.105*** (0.003)
$\log(Size)$	0.217*** (0.005)
$\log(Garage + 1)$	0.014*** (0.002)
$Renovate$	-0.017*** (0.001)
$\log(School)$	0.031*** (0.005)
$\log(Flood)$	0.028*** (0.001)
$\log(Fire)$	-0.007*** (0.002)
$\log(Heat)$	-0.208*** (0.007)
$\log(Wind)$	0.147*** (0.013)
$\log(Air)$	0.024*** (0.005)

$PropertyTypeDuplexTriplex$	0.060** (0.027)
$PropertyTypeMultiFamily$	-0.006 (0.006)
$PropertyTypeSingleFamily$	0.078*** (0.003)
$PropertyTypeTownhome$	-0.030*** (0.003)
baseline: Condos	
ZIP code fixed effects	Yes
N	284,749
Within R^2	0.6375

Note. Standard errors in parentheses are robust and clustered by property. * < 0.1, ** < 0.05, *** < 0.01.

B.2 Comparing with FairML Model

Our main results show that housing price discrimination is lower with GPT-generated prices compared with human-generated prices. In this appendix, we expand our analysis by considering machine learning models that are designed specifically to provide fairness in prediction results. Specifically, we utilize a similar set of information included in the GPT prompt, encode the categorical information using the leave-one-out method, and employ Optuna (Akiba et al. 2019) with the tree-structured Parzen estimator (TPE) algorithm (Watanabe 2023) and 3-fold cross-validation to search for the optimal hyperparameter set for 200 trails.

The development of our fairML includes two stages, involving both pre-process and in-process methods. First, although we do not directly include the discrimination factor in an ML model (i.e., whether the property is in a white-dominant area), the model's fairness can still be violated due to the associations between the discrimination factor and the model's variables (i.e., property characteristics and neighborhood information). As such, our first stage aims to mitigate these correlations, and we employ the CorrelationRemover, a pre-processing algorithm from the Fairlearn package developed by Microsoft (Bird et al. 2020). This algorithm performs a linear transformation on the model's variables, aiming to attenuate the correlations while minimizing the least-squares error between the original and transformed information. The extent of the linear projection (alpha) corresponds to the degree of correlation removal, and it is a crucial hyperparameter, influencing the trade-off between the model's fairness and its performance.

The second stage involves developing an ML model to achieve the highest model fairness and performance. We train the model with the transformed variables and select Extreme Gradient Boosting (XGB) as the algorithm, as it demonstrated the highest performance, considering both training time and mean absolute percentage error (MAPE). We define fairness as the absolute difference between MAPE for each group (i.e., white-dominant areas and not) as it reflects the disparity in the cost of utilizing a model between groups (Corbett-Davies et al. 2023). A value equal to zero indicates a (perfectly) fair model. Because our training aims to balance fairness and performance, we combine the value with the model's overall MAPE, considering the combined value as the model loss. We then set the objective function during the hyperparameter search to minimize the mean of this loss value across 3-fold. The results, reported in Table B.2, show that the bias is reduced more when housing prices are generated by the fairML model. However, the R^2 value of the fairML model is significantly lower than that of GenAI. This trade-off between fairness and accuracy is commonly observed in the literature (e.g., Fu et al. 2021, B. Bai et al. 2022a).

Table B.2: Price generated by Fair ML and ChatGPT

	$\log(Price)$ FairXGB	$\log(Price)$ GPT
AI baseline: human-generated prices	-0.650*** (0.002)	-0.149*** (0.001)
$MarketHighSupply$ baseline: low-supply areas	0.037*** (0.010)	0.061*** (0.011)
$MarketNewHouse$ baseline: old houses	0.041*** (0.002)	0.056*** (0.002)
$Underprivileged$ baseline: privileged areas	-0.170*** (0.005)	-0.227*** (0.005)
$AI \times Underprivileged$	0.124*** (0.006)	0.084*** (0.003)
$\log(Bed)$	-0.073*** (0.004)	-0.016*** (0.004)
$\log(Bath)$	0.228*** (0.003)	0.232*** (0.004)
$\log(Size)$	0.469*** (0.004)	0.597*** (0.004)
$\log(Garage + 1)$	0.043*** (0.002)	0.077*** (0.002)
$Renovate$	0.011*** (0.002)	0.033*** (0.002)
$\log(School)$	0.156*** (0.004)	0.237*** (0.005)
$\log(Flood)$	0.014*** (0.001)	0.011*** (0.001)
$\log(Fire)$	-0.018*** (0.002)	-0.025*** (0.002)
$\log(Heat)$	-0.103*** (0.007)	-0.103*** (0.008)

$\log(Wind)$	0.114*** (0.012)	0.135*** (0.014)
$\log(Air)$	0.035*** (0.004)	0.038*** (0.005)
$\log(Population)$	-0.048*** (0.003)	-0.061*** (0.003)
House type fixed effects	Yes	Yes
Location fixed effects	Yes	Yes
N	170,850	170,850
R^2	0.7577	0.8737

B.3 Traditional AI Models

In this appendix, we develop several supervised ML models based on the exact information included in the GPT prompt. For this analysis, we split 70% of the data into the training set and 30% into the test set.

We consider two sets of ML models. The first set includes commonly used ML models in the information systems literature (e.g., Liu et al. 2020, Wang et al. 2021, Xu et al. 2024, Yu et al. 2024) and the second comprises state-of-the-art algorithms. The first set consists of Random Forest (RF), Extreme Gradient Boosting (XGB), Light Gradient-Boosting Machine (LGBM), and Neural Network (NN). RF is an ensemble learning method that employs the bagging technique, leveraging several Decision Trees (DTs) to mitigate model's overfitting and enhance its performance (Breiman 2001). Both XGB and LGBM are gradient-boosted DT algorithms designed to enhance model performance and efficiency. However, they employ distinct tree growth strategies. While XGB adopts a depth-wise approach, prioritizing balanced tree structures and facilitating overfitting control, LGBM utilizes a leaf-wise approach, prioritizing the most impactful leaves for splitting, leading to higher accuracy and reduced loss (Chen and Guestrin 2016, Ke et al. 2017). Lastly, NN is a non-linear model that can learn complex relationships within data (Murtagh 1991).

The second set consists of Hybrid Regression (HYBRID), stacked generalization (STACK), and Categorical Boosting (CATBOOST). HYBRID is an ensemble technique that combines predictions from multiple models by equally weighting their outputs to achieve a final prediction (Zhang and Ma 2012). Our HYBRID model is constructed using the top three performing ML models from the first set. STACKING is another ensemble method that combines predictions from multiple models. However, based on their function, the models are categorized into two groups: base-learner, responsible for generating initial predictions, and meta-learner, which leverages these predictions as inputs to produce the final output (Zhang and Ma 2012). We employ the top performing ML model from the first set as the meta-learner. The remaining two models with the next best performance are the base models. CATBOOST is a contemporary gradient-boosting algorithm that employs decision trees designed to improve performance when handling categorical variables (Dorogush et al. 2018). Before training the ML models, we transform the categorical features into numerical representations. Since some categorical variables (i.e., ZIP code, city, state) are high dimensional, we employ the leave-one-out encoding method for the data transformation. This approach allows us to avoid sparse data and the curse of dimensionality, which usually result from the one-hot-encoding approach (i.e., the commonly used categorical encoding technique) and suffer the model performance. Furthermore, the method mitigates overfitting and data leakage by excluding the current sample during encoding (Vasques 2024). During each model's development, we use Optuna (Akiba et al. 2019), a cutting-edge hyperparameter optimization framework with a sophisticated trial-based search approach, to find the optimal hyperparameter set and utilize the tree-structured Parzen estimator (TPE) algorithm (Watanabe 2023), a Bayesian optimization method, as the sampling algorithm. The search is conducted for 200 trials, with the objective function of minimizing the mean of mean absolute percentage error (MAPE) across 3-fold. Tables B.3.1 and B.3.2 present the results, showing that the prices predicted by all ML models exhibit higher levels of discrimination compared to the prices generated by GPT as the coefficient of $SupervisedML \times Underprivileged$ is negative.

Table B.3.1: Price generated by off-the-shelf ML models

	$\log(Price)$ RF	$\log(Price)$ XGB	$\log(Price)$ LGBM	$\log(Price)$ NN
$SupervisedML$ baseline: GPT-generated prices	0.214*** (0.001)	0.155*** (0.001)	0.195*** (0.001)	0.134*** (0.004)
$MarketHighSupply$ baseline: low-supply areas	0.049*** (0.009)	0.053*** (0.009)	0.064*** (0.016)	0.014 (0.028)
$MarketNewHouse$ baseline: old houses	0.042*** (0.001)	0.052*** (0.001)	0.045*** (0.002)	0.032*** (0.004)
$Underprivileged$ baseline: privileged areas	-0.126*** (0.003)	-0.131*** (0.003)	-0.123*** (0.004)	-0.105*** (0.010)
$SupervisedML \times Underprivileged$	-0.090*** (0.002)	-0.082*** (0.002)	-0.078*** (0.004)	-0.200*** (0.013)
$\log(Bed)$	-0.0006 (0.003)	-0.008*** (0.003)	-0.039*** (0.004)	0.159*** (0.018)
$\log(Bath)$	0.213*** (0.002)	0.243*** (0.003)	0.245*** (0.005)	0.312*** (0.017)
$\log(Size)$	0.612*** (0.003)	0.586*** (0.003)	0.598*** (0.005)	0.521*** (0.019)
$\log(Garage + 1)$	0.057*** (0.001)	0.075*** (0.002)	0.068*** (0.003)	0.090*** (0.005)
$Renovate$	0.025*** (0.001)	0.035*** (0.001)	0.037*** (0.002)	0.039*** (0.004)
$\log(School)$	0.223*** (0.004)	0.238*** (0.004)	0.236*** (0.005)	0.365*** (0.015)
$\log(Flood)$	0.012*** (0.001)	0.010*** (0.001)	0.017*** (0.001)	0.010*** (0.004)
$\log(Fire)$	-0.022*** (0.001)	-0.031*** (0.001)	-0.037*** (0.002)	-0.057*** (0.004)
$\log(Heat)$	-0.046*** (0.005)	-0.071*** (0.005)	-0.073*** (0.007)	-0.117*** (0.019)
$\log(Wind)$	0.070*** (0.009)	0.099*** (0.009)	0.112*** (0.013)	0.083*** (0.030)
$\log(Air)$	0.030*** (0.004)	0.042*** (0.004)	0.037*** (0.005)	-0.016 (0.013)
$\log(Population)$	-0.045*** (0.002)	-0.047*** (0.002)	-0.048*** (0.003)	-0.022*** (0.008)
House type fixed effects	Yes	Yes	Yes	Yes
Location fixed effects	Yes	Yes	Yes	Yes
N	170,850	170,850	170,850	170,850
R^2	0.9351	0.9299	0.8611	0.5073

Table B.3.2: Price generated by advanced ML models

	$\log(Price)$ HYBRID	$\log(Price)$ STACK	$\log(Price)$ CATBOOST
$SupervisedML$ baseline: GPT-generated prices	0.187*** (0.0009)	0.204*** (0.0009)	0.195*** (0.001)
$MarketHighSupply$ baseline: low-supply areas	0.046*** (0.008)	0.046*** (0.008)	0.035*** (0.009)
$MarketNewHouse$ baseline: old houses	0.048*** (0.001)	0.044*** (0.001)	0.044*** (0.001)
$Underprivileged$ baseline: privileged areas	-0.124*** (0.003)	-0.119*** (0.003)	-0.118*** (0.003)
$SupervisedML \times Underprivileged$	-0.071*** (0.002)	-0.080*** (0.002)	-0.064*** (0.002)
$\log(Bed)$	0.015*** (0.003)	0.023*** (0.003)	-0.036*** (0.003)
$\log(Bath)$	0.247*** (0.002)	0.267*** (0.003)	0.249*** (0.003)
$\log(Size)$	0.559*** (0.003)	0.558*** (0.003)	0.611*** (0.003)
$\log(Garage + 1)$	0.062*** (0.001)	0.061*** (0.001)	0.069*** (0.002)
$Renovate$	0.033*** (0.001)	0.029*** (0.001)	0.038*** (0.001)
$\log(School)$	0.237*** (0.003)	0.240*** (0.003)	0.255*** (0.004)
$\log(Flood)$	0.012*** (0.0009)	0.012*** (0.0009)	0.017*** (0.001)
$\log(Fire)$	-0.027*** (0.001)	-0.023*** (0.001)	-0.033*** (0.001)
$\log(Heat)$	-0.062*** (0.005)	-0.053*** (0.005)	-0.103*** (0.005)
$\log(Wind)$	0.082*** (0.009)	0.070*** (0.008)	0.124*** (0.010)
$\log(Air)$	0.033*** (0.003)	0.030*** (0.003)	0.047*** (0.004)
$\log(Population)$	-0.047*** (0.002)	-0.046*** (0.002)	-0.054*** (0.002)
House type fixed effects	Yes	Yes	Yes
Location fixed effects	Yes	Yes	Yes
N	170,850	170,850	170,850
R^2	0.9393	0.9381	0.9240

C. Robustness Tests

In this appendix, we consider alternative specifications of our empirical analyses to demonstrate the robustness of our results.

C.1 Model without log-transformations

In our main analysis, we applied log transformation to all numerical variables that are power law distributed. In this robustness exercise, we consider a specification without log transformations. The results, reported in Table C.1, are consistent with our main results.

Table C.1: Model without log transformations.

	$Price_{ijk}$
AI baseline: human-generated prices	-138,713.3*** (1,258.3)
$MarketHighSupply$ baseline: low-supply areas	102,586.3*** (11,947.1)
$MarketNewHouse$ baseline: old houses	-24,926.7*** (4,238.2)
$Underprivileged$ baseline: privileged areas	-128,143.1*** (4,596.6)
$AI \times Underprivileged$	68,337.8*** (2,137.9)
Bed	-15,645.5* (8,999.1)
$Bath$	169,913.6*** (24,685.5)
$Size$	140.9*** (36.0)
$Garage$	94.4 (211.8)
$Renovate$	2,913.5 (2,198.3)
$School$	16,789.7*** (1,720.7)
$Flood$	4,763.4*** (509.6)
$Fire$	-4,029.2*** (824.4)
$Heat$	-18,143.8*** (4,264.6)
$Wind$	27,746.2*** (2,665.6)
Air	5,940.8*** (2,156.9)
$Population$	-71,932.6*** (4,114.2)
House type fixed effects	Yes
Location fixed effects	Yes
N	569,498
R^2	0.5309

Notes. Standard errors in parentheses are robust and clustered by property. * < 0.1, ** < 0.05, *** < 0.01.

C.2 Alternative Generative AI Models

Our main analysis relied on housing prices generated from GPT-4o. In this appendix, we consider alternative GenAI models to showcase that our results are not solely driven by our choice of GenAI model.

C.2.1 GPT 3.5

One of the common concerns for using GPT-4o in our analysis is that the model may generate prices by referencing actual listings from its training data, given the overlap in the observational periods and the model training time (i.e., the listings used in our data are between August and September 2023, and GPT-4o was trained by the data until October 2023). To address this concern, we consider an alternative GenAI model where we use housing prices generated by GPT-3.5 instead.

For efficiency purposes, we conduct this robustness test by randomly selecting 10% of houses from our dataset (i.e., 28,474 properties). For each property, we repeat the same process of using GPT-4o to generate a price, but we use GPT-3.5-turbo-0125 instead for this robustness exercise. We highlight that GPT-3.5 is trained on data obtained until September 2021. Therefore, there is no overlap between the trained data and observations in our dataset. Following that, to further ensure that our findings are not influenced by inflation, we also adjust the model-generated prices using the seasonally adjusted House Price Index (HPI) published by the Federal Housing Finance Agency (FHFA). The adjustment reflects the change in the index between Q3 2021 and Q3 2023, which corresponds to the time gap between the latest training data used in GenAI and the listings we collected. We then repeat our main analysis using GPT-3.5 generated prices and the prices with the inflation adjustment.

Table C.2.1: Housing prices from GPT 3.5

	$\log(Price_{ijk})$	$\log(Price_{ijk})$
	GPT-3.5	GPT-3.5 with Inflation Adjustment
AI baseline: human-generated prices	-0.378*** (0.001)	-0.206*** (0.001)
$MarketHighSupply$	0.041*** (0.006)	0.041*** (0.006)

baseline: low-supply areas		
$MarketNewHouse$ baseline: old houses	0.113*** (0.001)	0.113*** (0.001)
$Underprivileged$ baseline: privileged areas	-0.221*** (0.003)	-0.221*** (0.003)
$AI \times Underprivileged$	0.079*** (0.002)	0.079*** (0.002)
$\log(Bed)$	0.030*** (0.002)	0.030*** (0.002)
$\log(Bath)$	0.223*** (0.002)	0.223*** (0.002)
$\log(Size)$	0.587*** (0.003)	0.587*** (0.003)
$\log(Garage + 1)$	0.082*** (0.001)	0.082*** (0.001)
$Renovate$	0.036*** (0.001)	0.036*** (0.001)
$\log(School)$	0.209*** (0.003)	0.209*** (0.003)
$\log(Flood)$	0.013*** (0.001)	0.013*** (0.001)
$\log(Fire)$	-0.021*** (0.001)	-0.021*** (0.001)
$\log(Heat)$	-0.097*** (0.004)	-0.097*** (0.004)
$\log(Wind)$	0.129*** (0.008)	0.129*** (0.008)
$\log(Air)$	0.037*** (0.003)	0.037*** (0.003)
$\log(Population)$	-0.054*** (0.002)	-0.054*** (0.002)
House type fixed effects	Yes	Yes
Location fixed effects	Yes	Yes
N	569,498	569,498
R^2	0.8630	0.8564

Note. Standard errors in parentheses are robust and clustered by property. * < 0.1, ** < 0.05, *** < 0.01.

The results are reported in Table C.2.1. They are qualitatively similar to our main results. We note that we also replicate all results reported in the main paper and online appendix with GPT-3.5 and the vast majority of them are qualitatively similar to the results based on GPT-4o. These results are available upon request.

C.2.2 Gemini and Claude

GenAI models in the market exhibit significant heterogeneity in their underlying architectures and training framework. These differences can lead to varying responses to an identical question, raising concerns about whether our findings can be applicable across a spectrum of GenAI models. To address this potential limitation, we conduct a robustness test following the methodology outlined in C.2.1. Specifically, we utilize the same randomly selected 10% of properties from our dataset. We then replicate the price generation process with GPT-4o but instead utilize two other prominent GenAI models: Gemini and Claude.

We choose Gemini 1.5 Flash (gemini-1.5-flash-002) and Claude 3.5 Sonnet (claude-3-5-sonnet-20241022), which are the latest models as of the data collection date (October 28, 2024). Gemini was developed by Google, prioritizing factual correctness in its generated responses (Gemini Team 2024). Meanwhile, Claude was developed by Anthropic, placing significant emphasis on ethical considerations, particularly mitigating bias and discrimination (Y. Bai et al. 2022b). The results are reported in Table C.2.2, which shows qualitatively similar findings to our main analysis.

Table C.2.2: Housing prices from Gemini and Claude

	$\log(Price_{ijk})$ Gemini	$\log(Price_{ijk})$ Claude
AI baseline: human-generated prices	-0.154*** (0.003)	-0.178*** (0.002)
$MarketHighSupply$ baseline: low-supply areas	0.034* (0.019)	0.059*** (0.021)
$MarketNewHouse$ baseline: old houses	0.065*** (0.004)	0.057*** (0.004)
$Underprivileged$ baseline: privileged areas	-0.219*** (0.008)	-0.227*** (0.009)
$AI \times Underprivileged$	0.232*** (0.007)	0.077*** (0.006)
$\log(Bed)$	-0.061*** (0.007)	0.010 (0.008)
$\log(Bath)$	0.256*** (0.007)	0.243*** (0.008)
$\log(Size)$	0.555*** (0.008)	0.594*** (0.009)
$\log(Garage + 1)$	0.054*** (0.004)	0.063*** (0.004)
$Renovate$	0.019*** (0.003)	0.034*** (0.004)
$\log(School)$	0.172*** (0.008)	0.252*** (0.009)
$\log(Flood)$	0.013*** (0.002)	0.016*** (0.002)
$\log(Fire)$	-0.021*** (0.003)	-0.030*** (0.003)
$\log(Heat)$	-0.051*** (0.014)	-0.064*** (0.016)
$\log(Wind)$	0.143*** (0.024)	0.163*** (0.026)
$\log(Air)$	0.033*** (0.009)	0.041*** (0.010)
$\log(Population)$	-0.047*** (0.005)	-0.069*** (0.006)
House type fixed effects	Yes	Yes
Location fixed effects	Yes	Yes
N	56,948	56,948
R^2	0.8645	0.8789

Note. Standard errors in parentheses are robust and clustered by property. * < 0.1, ** < 0.05, *** < 0.01.

C.3 Alternative Discrimination Factors

Our primary analysis considers the majority racial composition within a neighborhood as a discriminatory factor and found that GenAI can mitigate housing price discrimination. To enhance the generalizability of our finding, this robustness test explores two alternative discriminatory factors. First, we examine the percentage of non-white residents relative to other racial groups within the neighborhood, identifying areas with a higher percentage as more underprivileged. Second, we consider household income within the neighborhood, classifying areas with an average income below the median income across all ZIP codes included in the dataset ($88,000) as the underprivileged group.

The results are presented in Table C.3, and they are qualitatively similar to our main result. Thus, we ensure that the conclusion of GenAI mitigating housing price discrimination does not solely rely on our choice of discrimination factor.

Table C.3: Impact of GenAI pricing on housing price discrimination

	Percentage of Minority	Income level
	$\log(Price_{ijk})$	$\log(Price_{ijk})$
AI baseline: human-generated prices	-0.196*** (0.001)	-0.155*** (0.0008)
$MarketHighSupply$ baseline: low-supply areas	0.036*** (0.006)	0.067*** (0.006)
$MarketNewHouse$ baseline: old houses	0.063*** (0.001)	0.054*** (0.001)
$Underprivileged$ baseline: privileged areas	-0.805*** (0.006)	-0.160*** (0.002)
$AI \times Underprivileged$	0.211*** (0.003)	0.053*** (0.001)
$\log(Bed)$	0.0006 (0.002)	-0.012*** (0.002)
$\log(Bath)$	0.232*** (0.002)	0.238*** (0.002)
$\log(Size)$	0.575*** (0.002)	0.581*** (0.003)
$\log(Garage + 1)$	0.075*** (0.001)	0.077*** (0.001)
$Renovate$	0.033*** (0.0009)	0.033*** (0.0009)
$\log(School)$	0.170*** (0.003)	0.234*** (0.003)
$\log(Flood)$	0.010*** (0.0007)	0.010*** (0.0007)
$\log(Fire)$	-0.023*** (0.0009)	-0.030*** (0.0009)
$\log(Heat)$	-0.104*** (0.004)	-0.112*** (0.004)
$\log(Wind)$	0.129*** (0.008)	0.127*** (0.008)
$\log(Air)$	0.041*** (0.003)	0.037*** (0.003)
$\log(Population)$	-0.042*** (0.002)	-0.063*** (0.002)
House type fixed effects	Yes	Yes
Location fixed effects	Yes	Yes
N	569,498	569,498
R^2	0.8690	0.8642

C.4 Alternative Data Collection Period

The primary data source of our study consists of housing prices between August and September 2023, which was after the release of GPT-3.5. While it is unlikely that the data we collected were affected by GenAI, given that GenAI only enabled the use of natural-language prompts to estimate housing prices after the release of GPT-3.5-turbo-0125 on January 25, 2024, we conduct a robustness test with alternative data collection period to further ensure the robustness of our findings by considering listing prices recorded prior to the public launch of ChatGPT on November 30, 2022. To ensure comparability between the current attributes and the historical listings, we restricted this analysis to the most recent historical listing price from 2020 for each property. In total, this dataset contains 24,377 properties with historical prices.

Our analysis is based on our main GenAI model (i.e., GPT-4o), whose most recent training data is from 2023. Thus, before conducting the analysis, we adjusted the historical prices for inflation to 2023 levels. Table C.4 presents the results of this robustness test, which are qualitatively similar to those of our main analysis.

Table C.4: Model with adjusted historical listing prices.

	$Price_{ijk}$
AI baseline: human-generated prices	-0.138*** (0.004)
$MarketHighSupply$ baseline: low-supply areas	0.102*** (0.026)
$MarketNewHouse$ baseline: old houses	0.067*** (0.005)
$Underprivileged$ baseline: privileged areas	-0.239*** (0.011)
$AI \times Underprivileged$	0.112*** (0.008)
$\log(Bed)$	-0.074*** (0.010)
$\log(Bath)$	0.228*** (0.010)
$\log(Size)$	0.672*** (0.010)
$\log(Garage + 1)$	0.086*** (0.005)
$Renovate$	0.011** (0.004)
$\log(School)$	0.239*** (0.011)
$\log(Flood)$	0.017*** (0.003)
$\log(Fire)$	-0.034*** (0.004)
$\log(Heat)$	-0.081*** (0.024)
$\log(Wind)$	0.159*** (0.031)
$\log(Air)$	0.050*** (0.011)

$\log(Population)$	-0.074*** (0.007)
House type fixed effects	Yes
Location fixed effects	Yes
N	48,732
R^2	0.8292

Notes. Standard errors in parentheses are robust and clustered by property. * < 0.1, ** < 0.05, *** < 0.01.

C.5 Additional Control Variables

In this appendix, we consider the local interest rate and income levels as additional control variables. In our analysis, we use the state-level mortgage rates reported by the Consumer Financial Protection Bureau. Theses interest rates are based on the median home purchase price with a 10 percent down payment and a credit score range of 700 to 719, reflecting a common homebuyer profile. For income levels, we use the average household income at the ZIP code level, obtained from U.S. Census Bureau. These variables are incorporated along with other relevant factors in the main analysis, both in the prompt and control covariates during the analysis. The results, reported in Table C.5, are qualitatively similar to our main results.

Table C.5: Additional control variables (mortgage rates and income levels)

	$\log(Price_{ijk})$
AI baseline: human-generated prices	-0.107*** (0.002)
$MarketHighSupply$ baseline: low-supply areas	0.026 (0.021)
$MarketNewHouse$ baseline: old houses	0.036*** (0.004)
$Underprivileged$ baseline: privileged areas	-0.126*** (0.009)
$AI \times Underprivileged$	0.079*** (0.006)
Control variables	$\log(Bed)$, $\log(Bath)$, $\log(Size)$, $\log(Garage+1)$, $Renovate$, $\log(School)$, $\log(Flood)$, $\log(Fire)$, $\log(Heat)$, $\log(Wind)$, $\log(Air)$, $\log(Population)$, $\log(MortgageRate)$, $\log(MeanIncome)$
Fixed effects	House type, Location
N	569,498
R^2	0.8896

Note. Standard errors in parentheses are robust and clustered by property. * < 0.1, ** < 0.05, *** < 0.01.

D. Additional Figures

In addition, to the main figure we develop in Section 4.3.1 to explore the potential central tendency behavior, we create additional figures that plot the distributions of AI-generated and human-generated prices for different segments, including high-populated vs. low-populated areas, old houses vs. new houses, and low-supply vs. high-supply areas. In the majority of cases, we observe that the variance of AI-generated prices is notably higher than that of human-generated prices, hence suggesting that central tendency is unlikely to be the mechanism explaining our study's results.

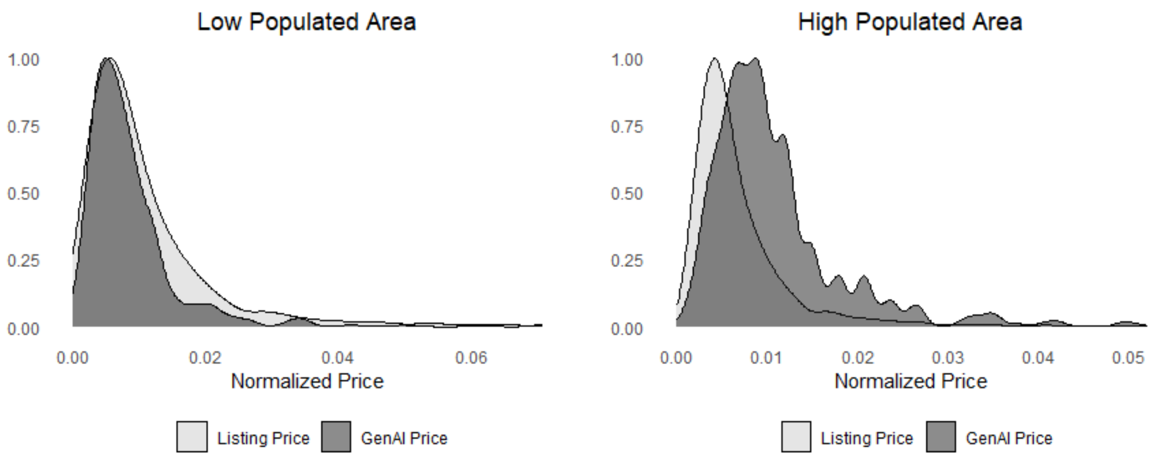

Figure D.1: Distribution of AI- and human-generated prices (low- vs. high-populated areas)

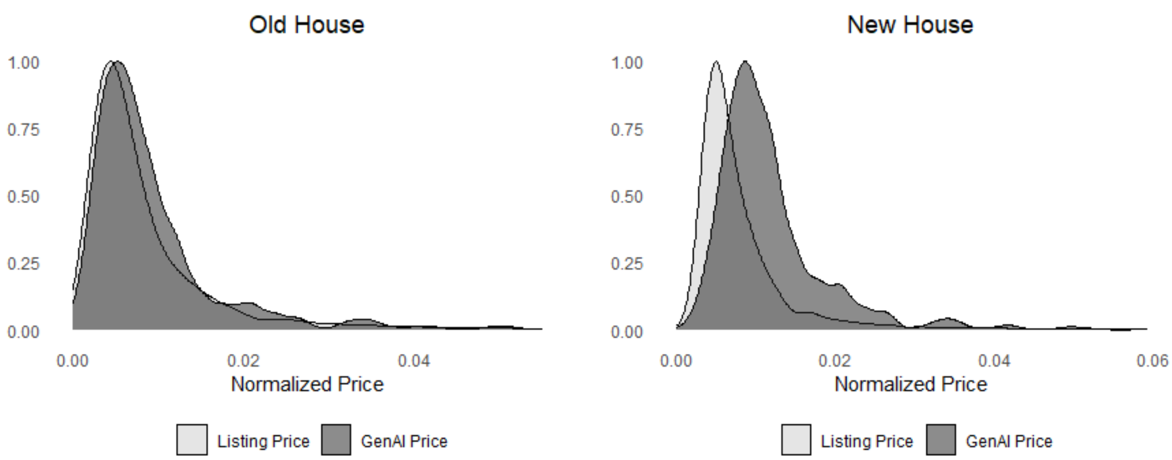

Figure D.2: Distribution of AI- and human-generated prices (old vs. new houses)

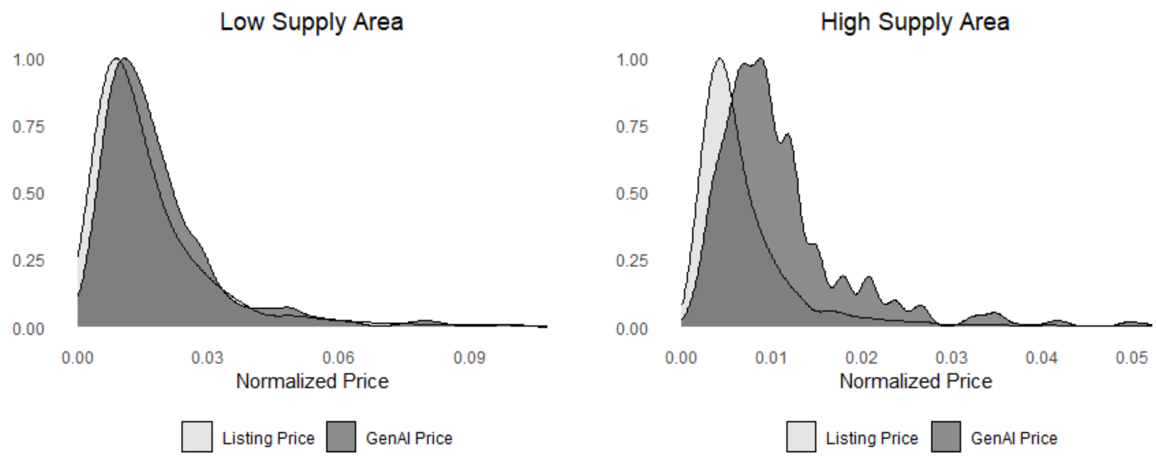

Figure D.3: Distribution of AI- and human-generated prices (low- vs. high-supply areas)

E. The Use of GenAI in Real Estate Industry

Our study provides evidence on changes in housing price discrimination toward marginalized consumers when prices are generated by GenAI in the experimental setting. Naturally, this raises important questions regarding whether real estate agents actually use GenAI to generate housing prices and how AI-generated prices influence homeowners' perceptions of the market value of their properties in real-world settings. In this section, we report findings from informal interviews with real estate agents, as well as two small-scale online surveys involving self-reported real estate agents and homeowners, to provide limited but supportive evidence regarding the use of GenAI in the real estate industry.

E.1 Real Estate Agents and GenAI

E.1.1 Informal Interview with Real Estate Agents

To understand how real estate agents use GenAI for pricing purposes, we engaged in a conversation with a prominent real estate agency in North America. We have leveraged this collaboration to learn the factors that impact real estate agents' pricing decisions. These factors include ZIP code (aka "location location location"), city, state, housing age, number of bedrooms, number of bathrooms, house size, garage space, property type, school rating, flood risk, wildfire risk, heat risk, wind risk, air pollution risk, and whether it was recently renovated. We have also learned that real estate brokers have increasingly relied on GenAI tools to help set the listed price of the properties they handle. The key attractiveness of GenAI tools (as opposed to traditional AI models) is the ease of use and trust in the technology since the vast majority of agents already use GenAI tools in their daily lives. As such, the use of GenAI to generate housing prices is often viewed as a data-driven approach that leverages historical data and state-of-the-art technologies. Generally, the real estate agents feed the details of the properties, including the property and neighborhood characteristics listed above, to the GenAI tool and ask the tool to estimate (or recommend) the price of the property. The generated prices are then reviewed and adjusted as necessary. Our conversation revealed that many real estate agents view housing prices generated by GenAI tools as being reasonably accurate (i.e., in line with the prices that they would have generated themselves), except for extreme outlier cases. As such, real estate agents tend to view GenAI as a productivity tool that allows them to price tens of properties in a single click with limited technical skills. Our separate conversations with several independent real estate agents also revealed that they tend to rely heavily on GenAI to price properties in areas that they are not familiar with.

E.1.2 Online Survey

We administered a short survey on the Prolific platform. Regarding the sample size, we followed Moser and Korstjens (2018), who recommend a minimum sample size of 30 for content analysis. To align with the context of our study, eligible participants were limited to real estate agents based in the United States. The survey begins by collecting information on participants' characteristics, including years of professional experience and familiarity with generative AI tools. Table E.1.1 presents descriptive statistics summarizing these characteristics. Overall, the participants are relatively early-career real estate agents and are generally familiar with generative AI tools, aligning with our previous, anecdotal interviews with real estate agents from the partner company.

Table E.1.1: Characteristics of participants (real estate agents)

	N = 30
How long have you been working as a real estate agent?	
Less than 1 year	10 (33.33%)
1-4 years	14 (46.67%)

5-10 years	3 (10.00%)
More than 10 years	3 (10.00%)
How familiar are you with Generative AI tools like ChatGPT?	
Not familiar at all (I have never used these tools)	0 (0.00%)
Slightly familiar (I have heard of them but rarely use them)	6 (20.00%)
Moderately familiar (I have used them a few times)	12 (40.00%)
Very familiar (I use them regularly)	12 (40.00%)

In the survey, we asked participants for their opinions regarding the potential use of GenAI and its generated prices for listed properties. Participants were first asked, "Which tasks do you think generative AI tools can help with in your day-to-day real estate work?" We provided several predefined options and asked them to select all that applied. The options included writing property descriptions for listings, drafting emails or client communications, estimating asking prices for listed properties, and conducting market research or trend analysis. An open text box was also provided for additional responses. Figure E.1.1 presents the results, indicating that real estate agents primarily use GenAI for writing tasks (i.e., property descriptions, emails, and client communications). Notably, half of the participants reported that they use GenAI tools for pricing recommendations, also aligning with our previous interviews with real estate agents from the partner company.

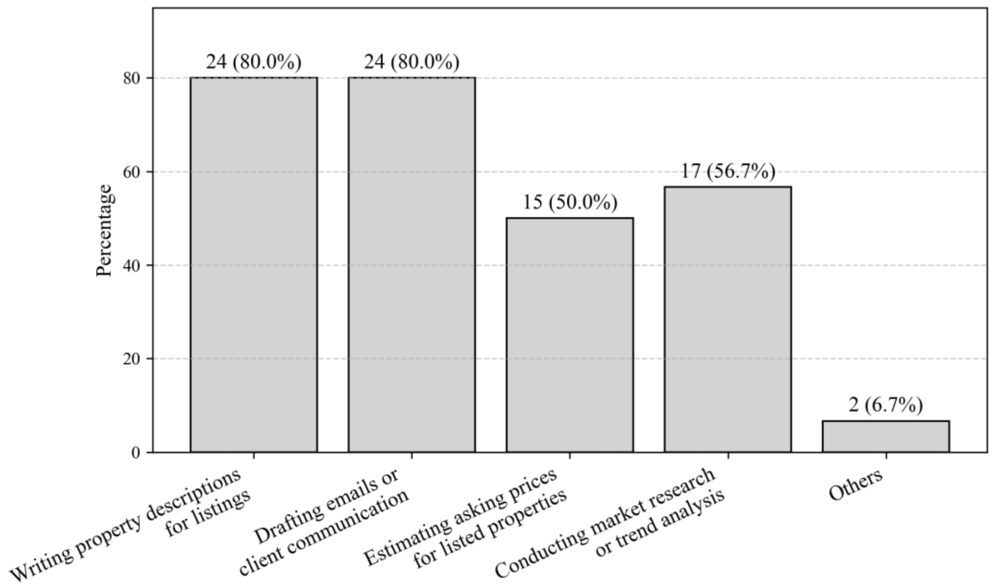

Figure E.1.1: Potential applications of GenAI in real estate tasks

Next, participants were asked two open-ended questions regarding their opinions about GenAI-generated prices: (1) When and why are you likely to use generative AI tools to guide setting the asking price for properties? and (2) What do you think about the price suggested by a generative AI tool like ChatGPT?

To formally analyze the responses, one co-author served as the primary coder and performed the iterative coding process, including developing a coding scheme and double-coding to ensure reliability (Boréus and Bergström 2017). The same coding scheme was then applied by a second coder, and any differences in interpretation were resolved through discussion until consensus was reached. Table E.1.2 presents the results of our qualitative analysis.

Table E.1.2: Qualitative analysis for open-ended questions (real estate agents)

Theme (N, %)	Example
When and why are you likely to use Generative AI tools to guide setting the asking selling price for properties?	
Market Analysis (13, 43.33)	• To compare home prices in the area im selling. (P24) • I mostly use generative AI applications when I am preparing a comparative market analysis (CMA) or determining the starting asking price of a property… (P53-1)
Workload & Time Efficiency (11, 36.67)	• I'd use Generative AI for quick pricing estimates and to save time analyzing market trends. (P39)

	• If I have a lot of properties to work on for the day, I could use Generative AI to help gather data from other houses in the area as well as market trends to set a proper asking price. (P9)
Pricing in Unfamiliar Settings (7, 23.33)	• To help set a property's asking price when the market is unpredictable or the home has unique features that make it hard to compare. (P40) • I might try using Generative AI tools to guide me in setting asking selling prices for homes in areas that I am not familiar with. (P48)
Information Validation (2, 6.67)	• I use it primarily to ensure the data I have for current market pricing for this area is accurate and up to date. (P54) • The extent of price estimation is rather limited, other than using it as a tool to check logic and assumptions. It's a good tool for compiling mass amounts of data and being a reasonable sanity check... (P37)
Others (2, 6.66)	• I may ask for a range, but since each home is unique, it is up to the professional to determine the final price. (P21) • I haven't used it for this purpose yet, but it seems like a good idea. (P42)
What do you think about the price suggested by a Generative AI tool like ChatGPT?	
Reliable (15, 50.00)	• I think the pricing is fair as it does a search of all prices and recommends the best. (P46) • I think it will be more effective compared to whatever price i can suggest, considering it will consider wider range of factor. (P55)
Good Baseline or Starting Point (12, 40.00)	• From my perspective, I would view the price suggested by a Generative AI tool like ChatGPT as a strong starting point rather than a final decision. I think it's valuable because it pulls insights from large datasets, market trends, and property features that I might not immediately see on my own. At the same time, I believe it's important to validate AI-generated suggestions with local market knowledge, human expertise, and real-world context... (P34) • They can be a useful starting point. They can quickly analyze market trends, property details and compare sales to make a fair estimate, I see it as a guide rather than a final answer. (P33)
Unreliable (3, 10.00)	• If its based on real time and accurate data, i would trust that information. However, its not always accurate and that causes issues to rely on it solely. (P40) • Unique Properties: For custom-built homes, historic estates, or properties in rural areas with limited comparable sales, the AI's accuracy can decrease. (P29)

Based on the responses, we observe that real estate agents recognize the potential use of GenAI for pricing in various scenarios, primarily for market analysis, when efficiency is critical, and when estimating prices in unfamiliar settings. In addition, the majority of participants consider GenAI-generated prices to be reliable. Specifically, only about 10 percent of participants express skepticism regarding the prices suggested by GenAI. Overall, the survey results provide strong evidence that real estate agents tend to use GenAI in the pricing process and that they generally find GenAI-generated housing prices reliable.

E.2 Homeowners and GenAI

We conducted an additional online experiment to investigate whether AI-generated prices affect homeowners' final decisions when setting listing prices. Assuming a medium effect size of 0.3 and an alpha level of 0.05 for a two-tailed paired t-test, G*Power (Faul et al. 2007) recommends a minimum sample size of 119. We therefore recruited 120 participants on the Prolific platform for this experiment.

Before the experiment, we asked the participants three general questions: (1) their familiarity with GenAI, (2) their experience in selling real estate property, and (3) the resources they find helpful in determining an asking price. For the third question, we provided predefined options and asked participants to select all that applied. The options included researching recent sales of comparable properties themselves, consulting a real estate agent, consulting a generative AI tool (e.g., ChatGPT, Gemini, Claude), paying for a professional appraisal, and seeking opinions from friends, family, or neighbors. We also provided an open text box to allow participants to offer additional input not covered by the predefined options. Table E.2.1 and Figure E.2.1 present the results from these questions. We find that nearly all participants have some experience with GenAI tools, and about half have never sold a property before. Meanwhile, several participants consider GenAI a useful resource for assisting with property sales.

Table E.2.1: Characteristics of participants (homeowners)

	N_s = 120
How familiar are you with Generative AI tools like ChatGPT?	
Very familiar (I use them regularly)	78 (65.00%)
Moderately familiar (I have used them a few times)	31 (25.83%)
Slightly familiar (I have heard of them but rarely use them)	10 (8.33%)
Not familiar at all (I have never used these tools)	1 (0.83%)
How experienced are you with the process of selling a real estate property?	
Very experienced (I have sold multiple properties)	18 (15.00%)
Somewhat experienced (I have sold one property before, or I am in the process of selling one)	54 (45.00%)
Not experienced (I have never sold a property)	48 (40.00%)

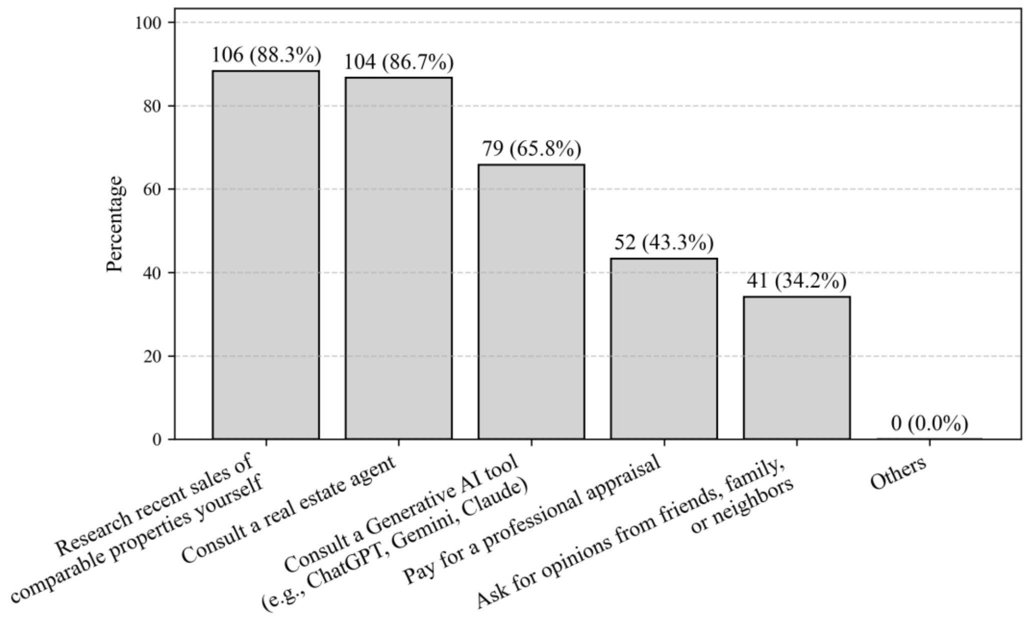

Figure E.2.1: Resources perceived as useful for pricing

Next, the participants entered the actual experiment. We began by presenting each participant with information about a property, prefaced with an instruction to assume that it was their own real estate property. The information presented was consistent with that included in the prompt used to request a price estimate from GPT in our study. An example is shown below:

Suppose that you own a real estate property with the following characteristics:

Property Type	*: Condos*
Zip code	*: 98274*
City	*: Mount Vernon*
State	*: WA*
House Age	*: 21 years*
Renovation	*: No*
Bed	*: 2*
Bath	*: 2*
Sqft	*: 855*
Garage Space	*: None*
Mean School Rating (Out of 10)	*: 2.85*
Environmental Risk (Out of 10)	
Flood	*: 1*
Wildfire	*: 3*
Heat	*: 2*
Windstorms	*: 1*
Air Pollution	*: 5*

Following that, we asked the participants the two main questions of this experiment. First, we asked them to estimate the listing price based solely on the property information provided. Next, we presented a scenario in which the participant consults a GenAI tool (e.g., ChatGPT, Gemini, Claude) for a price estimate. We explained that the tool analyzes the property details and provides a recommended listing price (e.g., $320,000 for the example above). After receiving this information, the participants were asked to set the final listing price for the property.

Each participant completed this sequence of questions for a total of five properties (referred to as five tasks), representing each property type in our empirical data (i.e., condo, duplex/triplex, multi-family, single-family, and townhome). All property information and GenAI-generated prices were randomly selected from our empirical dataset. We also note that all the participants received the exact same set of property descriptions and GenAI-generated prices, although the order in which the properties were presented was randomized.

We analyze the data from this online experiment using a paired t-test, comparing listing prices before and after the participants are exposed to AI-generated prices. To account for differences in the direction of the price discrepancy, we separate the data based on whether participants' initial estimates are higher or lower than the GenAI-generated prices. Table E.2.2 presents the results, showing that the AI price suggestions influence homeowners' final pricing decisions regardless of the direction of the discrepancy. Specifically, homeowners significantly adjust their estimates toward the AI's suggestion. They raise their final listing prices when their initial estimates are lower than the GenAI-generated price, and they lower their final listing prices when their initial estimates are higher.

Table E.2.2: Experiment results for the homeowners

Price Comparison	N	$\mu_{Initial}$	μ_{Final}	t-statistics	H0: $\mu_{Initial} = \mu_{Final}$
Initial <= GenAI	260	397,269.43	702,841.92	-7.332***	Reject
Initial > GenAI	340	1,177,227.94	414,744.12	1.888*	Reject

Notes * < 0.1, ** < 0.05, *** < 0.01.

In summary, the results from the surveys of self-reported real estate agents in Section E.1.2 and homeowners in Section E.2 provide supportive evidence regarding the use of GenAI in the real estate industry. Nevertheless, it is important to acknowledge the limitations of these surveys, as they rely on self-reported occupations and involve relatively small sample sizes. Accordingly, these findings should be interpreted as limited but supportive evidence and should not be overly generalized as validation of our results in the real-world housing market.

References

Akiba T, Sano S, Yanase T, Ohta T, Koyama M (2019) Optuna: A next-generation hyperparameter optimization framework. *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2623-2631.

Alto V (2024) *Building LLM Powered Applications* (Packt Publishing, Birmingham, UK).

Bai B, Dai H, Zhang DJ, Zhang F, Hu H (2022a) The impacts of algorithmic work assignment on fairness perceptions and productivity: Evidence from field experiments. *Manufacturing & Service Operations Management* 24(6):3060-3078.

Bai Y, Kadavath S, Kundu S, Askell A, Kernion J, Jones A, Chen A, Goldie A, Mirhoseini A, McKinnon C (2022b) Constitutional AI: Harmlessness from AI feedback. Preprint

Bird S, Dudík M, Edgar R, Horn B, Lutz R, Milan V, Sameki M, Wallach H, Walker K (2020) Fairlearn: A toolkit for assessing and improving fairness in AI. Preprint.

Breiman L (2001) Random forests. *Machine learning* 45:5-32.

Chen T, Guestrin C (2016) Xgboost: A scalable tree boosting system. *The 22nd ACM sigkdd international conference on knowledge discovery and data mining*, 785-794.

Corbett-Davies S, Gaebler JD, Nilforoshan H, Shroff R, Goel S (2023) The measure and mismeasure of fairness. *The Journal of Machine Learning Research* 24(1):14730-14846.

Dorogush AV, Ershov V, Gulin A (2018) CatBoost: gradient boosting with categorical features support. Preprint

Faul F, Erdfelder E, Lang A-G, Buchner A (2007) G* Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences. *Behavior research methods* 39(2):175-191.

Fu R, Huang Y, Singh PV (2021) Crowds, lending, machine, and bias. *Information Systems Research* 32(1):72-92.

Gemini Team (2024) Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. Preprint

Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu T-Y (2017) Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems* 30.

Liu Y, Pant G, Sheng OR (2020) Predicting labor market competition: Leveraging interfirm network and employee skills. *Information Systems Research* 31(4):1443-1466.

Moser A, Korstjens I (2018) Series: Practical guidance to qualitative research. Part 3: Sampling, data collection and analysis. *European journal of general practice* 24(1):9-18.

Murtagh F (1991) Multilayer perceptrons for classification and regression. *Neurocomputing* 2(5-6):183-197.

Ozili PK (2023) The acceptable R-square in empirical modelling for social science research. *Social research methodology and publishing results: A guide to non-native English speakers* (IGI global), 134-143.
Vasques X (2024) Feature Engineering Techniques in Machine Learning. *Machine Learning Theory and Applications* (Wiley), 59.
Wang G, Chen G, Zhao H, Zhang F, Yang S, Lu T (2021) Leveraging Multisource Heterogeneous Data for Financial Risk Prediction: A Novel Hybrid-Strategy-Based Self-Adaptive Method. *MIS Quarterly* 45(4).
Watanabe S (2023) Tree-structured parzen estimator: Understanding its algorithm components and their roles for better empirical performance. Preprint.
Xu Y, Ghose A, Xiao B (2024) Mobile payment adoption: An empirical investigation of Alipay. *Information Systems Research* 35(2):807-828.
Yu Y, Tan X, Tan Y (2024) Understanding Volunteer Crowdsourcing from a Multiplex Perspective. *Information Systems Research*.
Zhang C, Ma Y (2012) *Ensemble machine learning: methods and applications* (Springer).