Land Value Tax Would Solve This

Estimating Land Value for Improved Property Taxes

Todd Nief
Advanced Data Analytics - MPCS 53112
Code: https://github.com/toddnief/land-value-tax-would-solve-this

December 2022

Abstract

In most jurisdictions, property is taxed based upon recent sale prices of comparable units. However, there is an economic case to be made for taxing land value at close to 100% while not taxing buildings or improvements *at all*. The land value tax is favored by both right and left-leaning economists as a tax that does not negatively impact economic activity while also enabling redistribution of wealth to neighboring communities. A land value tax would destroy many of the perverse incentives under the current property tax regime for landlords to speculatively hold high-value lots, often in an underdeveloped or vacant state, in hopes of making money on a future sale. It would also incentivize landlords to increase the value of their buildings resulting in more housing density in urban centers and decreasing sprawl. The economic arguments for land value tax are strong, but is it realistic to assess land value separately from property value? In this project, I recreate the non-parametric kernel regression land value estimation method from Kolbe, et al's paper "Identifying Berlin's land value map using Adaptive Weights Smoothing" and use this estimated land value in a two-step log-linear regression to estimate land value and model sale price. I use a data set from the City of Philadelphia's tax year 2023 property tax and assessment data. The results for the two-part regression were close to meeting industry standards for property tax assessments and show promise for creating a split tax that separates land value and building value.

1 Introduction

Henry George, a 19th century economist, popularized a single tax to end all taxes — a 100% tax on the value of land itself. The economic reasoning for this tax is well-accepted by academic economists on both the left and the right. Venerated University of Chicago Maroon and free-market enthusiast Milton Friedman has called a land value tax "the least bad tax," and left-leaning economists also favor land value tax as a model for wealth redistribution.

Land is unique among taxable assets: the supply of land is fixed. As such, taxing land does not create "deadweight loss" (economic activity benefiting both the buyer and the seller) since the supply cannot adjust in response to market demand - landlords will not stop renting if land is taxed at a higher rate. In fact, they will be incentivized to create more value on their property in order to get a financial benefit from ownership. Additionally, the value of the land is derived

almost entirely from the surrounding community rather than from specific activities that the land owner engages in. Landlords reap financial benefit as the community around their property improves. A land value tax would enable the community to recapture some of this value rather than allowing landlords to capture all of it.

The current property tax paradigm also creates perverse incentives favoring speculation and long-term vacancies. Landlords can make money by sitting on undeveloped properties or converting them into low value-add uses (like parking lots) in hopes of future appreciation of real estate values. Landlords will also leave commercial or residential properties vacant or in disrepair for long periods of time in hopes of selling a highly appreciated property in the future. Making improvements in the short term is a cash expense that may not pay off. Additionally, since property taxes are based on the value of *buildings*, landlords are punished for making improvements to properties - there are also sometimes tax incentives and write-offs for vacant properties!

Finally, since the value of land continues to increase as the surrounding community becomes more and more active, the percentage of business income (for commercial real estate) or a renter's paycheck (for individuals) that landlords can capture increases over time. The incentives to speculate reduce the number of commercial and residential properties on the market for rental, driving the price up for everyone. If landlords had to make money by offering value through their buildings, the supply and quality of available units would increase.

If we accept that a land value tax would solve some of these problems, how do we move from the current property tax paradigm to one that taxes land value directly? Rather than switching immediately to a land value tax, implementing "split assessments" with a land value line item and an improvements line item is more realistic.

However, most county assessors offices, like many civil servants, are badly overworked and underpaid. Even though many county assessors are open to creating a split tax, they do not have the staffing power and flexibility to separately assess land value and building improvements. By leveraging statistical techniques to predict land value, assessors can compare algorithmic predictions with the outputs of their current assessments to implement split property taxes. Additionally, it's important for property tax assessments to be human-interpretable. Techniques like linear regression that show a clear relationship between property features and assessed value are preferable, while black box techniques like neural networks that can learn confusing, non-linear relationships are likely to cause political problems.

1.1 Two-Part Regression for Land Value Estimation

County assessors typically assess properties by modeling historical real estate transactions. All real estate sales are recorded, so assessors model the sale price of a specific parcel of land based upon the features of that parcel. These features include things like total area of the parcel, estimated consruction date, grade of the exterior quality of the building, etc. They then evaluate model performance using a ratio of assessed to actual value and real estate specific statistics like the coefficient of dispersion and price-related differential (see Section 3.5 for more detail). Outliers are manually adjusted, and models are tweaked to meet industry standards.

In this work, I split the modeling of sale price into two steps. First, I use a non-parametric kernel regression model to estimate land values for all properties in the City of Philadelphia using sale history for vacant lot transactions. Then, I subtract this land value from sale prices and use log-linear regression to predict the remaining

sale price. This model performs close to industry standards for residential buildings in Philadelphia, so it shows promise for creating a split property tax system.

2 Related Work

There is a variety of work on real estate sale price prediction, but I focused my literature review on finding work on vacant land value assessment and property value prediction using geographic features.

Kolbe, et al. performed non-parametric kernel regression using data from Berlin in "Identifying Berlin's land value map using Adaptive Weights Smoothing." My work is based largely on this paper. This paper shows that the adaptive weights smoothing algorithm predicts land values similar to expert assessments in Berlin. In their follow up work "Land value appraisal using statistical methods", Kolbe et al. revisit the adaptive weights smoothing algorithm with updated kernel functions. They also propose a two-step method for estimating land values in urban centers lacking sufficient vacant land transactions. They calculate the residuals for regression of sale price using only building features, then use the adaptive weights smoothing procedure to get an estimator for the residuals for each geographic area.

Helbing, et al. apply the same adaptive weights smoothing procedure to arable land in "Estimating location values of agricultural land." They also show results comparable to expert performance, and show that experts underweight land value compared to their algorithm.

Yalpir & Unel have a multiple regression model in "Use of Spatial Analysis Methods in Land Appraisal; Konya Example" that uses GIS data. They use real estate transaction data from Konya, Turkey then overlay this with GIS data. They then calculate distance to amenities like health facilities, airports, bus stops, etc. They use each of these variables in a multiple regression analysis along with some building features, and obtain a model that performs well with an R^2 score of .85.

In "Estimating a Price Surface for Vacant Land in an Urban Area," Colwell & Munneke use a different form of nonparametric regression to estimate land values in Chicago. Similar to the adaptive weights smoothing technique, they collect sales data for vacant lots in Chicago and the surrounding suburbs. They estimate a price surface by using barycentric coordinates relative to changes in price for vacant land transactions - this creates a topology defined by points where there are significant changes in land value.

3 Estimating Land Values and Sale Prices in Philadelphia

I used sale price and property tax assessment data for taxes due in 2023 from https://www.opendataphilly.org/. Additionally, I used data provided by Lars Doucet of https://www.gameofrent.com/ that joins the Philadelphia property tax data with information on the nearest park, the elementary school district, and the distance to Philadelphia's town hall (a proxy for distance to downtown). These joins were created using other data from Open Data Philly.

This work has three major phases: data-cleaning, non-parametric kernel estimation of land value, and log-linear regression on sale price. The City of Philadelphia data includes over 570,000 properties and was split into a training set of 465,120

and a testing set of 116,281. The testing set was held out until the final models were evaluated.

3.1 Data Cleaning Pipeline

The data from the City of Philadelphia required significant cleaning to prepare it for regression. There were several feature columns with incorrectly or inconsistently coded values (for example Boolean fields containing 0, 1, Y, N), mixed datatypes (integers and strings), and several properties with missing values.

To handle missing values, I excluded any examples that were missing location data or total area data. If an example was missing target data for sale price, I did not use it in the regression process. For categorical values, I treated missing variables themselves as a category. If a property was missing a boolean value for "Central Air", I encoded it in a separate category as "Central Air Missing." One can imagine a situation where missing a value for something like building inspection would include important information about a property's value - more prominent and valuable properties are probably more likely to be consistently inspected, for example. All categorical variables were one-hot encoded without dropping features.

The documentation for Philadelphia's assessment methodology mentioned an algorithm that adjusted sale price based upon overall market performance at the time of sale — i.e. sales that occurred while the real estate market was high were adjusted downward and sales that occurred while the real estate market was low were adjusted upward. I couldn't find the methodology for these adjustments, so I did not perform any sale price adjustment. The Philadelphia data only included sale price for the previous five years, so it's reasonable to assume that sale prices are at least "in the ballpark" of contemporary sale prices.

There were several features that were encoded as continuous variables like number of off-street parking spots, number of fireplaces, and stories in the building. However, I binned all continuous variables other than the target (sale price) and total area, distance to town hall, and distance to nearest park. Binning most of the continuous variables yielded large improvements in cross-validation. I made judgment calls on how to bin things like number of stories, number of garage parking spaces, etc. The binning information is available in the project Jupyter notebook.

I performed log-linear regression on sale price per square foot rather than attempting to model sale price directly. This is in alignment with the City of Philadelphia's methodology.

I also created geographic bins in order to estimate land value for each property. To create the geographic bins, I found a bounding box for all examples in the data set. I then experimented with different bin sizes using cross-validation on regression results and visual inspection of the average land value map. Bin size is an important hyperparameter in the accuracy of the kernel land value estimation, since bins that contain only a few transactions can have large outliers in terms of average price. For example, small bins may result in a bin in the heart of downtown Philadelphia that only has one particularly cheap vacant land transaction. This bin then has an average sale price well below its neighbors — this causes problems in the kernel estimation stage since land values are iteratively estimated using a weighted average of their neighbors. Based upon cross-validation results and inspection of outliers, I used a bin size of 1000m x 1000m.

3.2 Kernel Estimation

I estimated land values throughout the City of Philadelphia using nonparametric regression.

The nonparametric regression process used is called adaptive weights smoothing and is derived from Kolbe et al.'s paper "Identifying Berlin's land value map using Adaptive Weights Smoothing." This method assumes that land values of adjacent properties are mostly similar and that changes in land value are mostly smooth.

This method uses an iterative estimation process based upon two kernels: a distance kernel and a leveling kernel. The basic intuition is that land that is both nearby and of a similar value is a good estimator for the land value of a specific parcel.

The adaptive weighted smoothing algorithm is run on the geographic bins described in the previous section. To help with intuition for this process, here are the steps in order:

1. Calculate the distance and level kernels for each bin relative to all other bins
2. Multiply the distance and leveling kernels to get a weight for each bin relative to each other bin
3. Calculate a weighted average of sale prices for each bin (using the multiplied kernels as the weights)
4. Iteratively repeat this process with increasing distance bandwidths using the previous land value estimates as input into the next step

First, we will discuss the distance and leveling kernels. Both kernels use the Epanechnikov kernel.

$$K(u) = \frac{3}{4}(1 - u^2) \quad \text{for } |u| \leq 1, \quad 0 \text{ otherwise}$$

For the distance kernel, u is the Manhattan distance (in bin length) between the two bins divided by the distance bandwidth:

$$u_{distance} = \frac{|x_i - x_j| + |y_i - y_j|}{h}$$

The leveling kernel is more complex.

$$u^k_{level} = \frac{n_i^{k-1}}{\lambda} \left(\frac{\hat{\theta}_i^{k-1} - \hat{\theta}_j^{k-1}}{\sqrt{\hat{\sigma}^{k-1\ 2}_{theta}}} \right)^2$$

In this equation, the superscript k represents the iteration number. The $\hat{\theta}^{k-1}$ terms are the estimated land values at the previous iteration step $t-1$, the $\hat{\sigma}$ term is standard deviation of the land values at the previous iteration, the n_i^{k-1} term is the number of bins used in the previous iterations estimate for bin i, and λ is the leveling bandwidth.

The leveling kernel can be thought of as a statistical test as to whether or not land values are close in price. Just as the distance kernel weights land values based upon how close the geographic bins are, the leveling kernel weights land values based upon how close in price they are. As the bandwidth for the distance kernel increases on each subsequent iteration, the leveling kernel ensures that only those bins with

similar land value impact the estimate of land value. To get some intuition for the leveling kernel, you might imagine that properties on the outskirts of opposite ends of a city should potentially impact each other's weighted average based upon having similar land values - even though they are far apart in distance. In this case, the leveling kernel corresponds to an intuition something like this: "Land prices in the suburbs of Philadelphia that are close in value are good estimators for each other - even if they are far apart in distance. However, vacant land in a part of the city that is very cheap for local reasons is not a good estimator for nearby land in a bustling commercial district even though it is geographically close."

To derive the leveling kernel, we assume as a null hypothesis that the land value in two geographic bins is the same. That is:

$$H_0 : \theta(\text{lat}_i, \text{lon}_i) = \theta(\text{lat}_j, \text{lon}_j)$$

We model our true land value for location i as:

$$y_i = \theta(\text{lat}_i, \text{lon}_i) + \epsilon_i$$

So, if the null hypothesis holds, we have:

$$\begin{aligned} \theta_i - \theta_j &\sim N(0, \sigma^2) \\ \frac{\theta_i - \theta_j}{\sqrt{2\sigma^2}} &\sim N(0,1) \\ \left(\frac{\theta_i - \theta_j}{\sqrt{2\sigma^2}}\right)^2 &\sim \chi^2(0,1) \end{aligned}$$

We consider λ (the leveling bandwidth) to be the critical value at which we reject the null hypothesis. We use the following to conduct our hypothesis test:

$$\frac{n_i}{\lambda}\left(\frac{\hat{\theta}_i - \hat{\theta}_j}{\sqrt{\hat{\sigma}^2}}\right)^2$$

If this value is greater than one, then we reject the null hypothesis that θ_i equals θ_j at our significance threshold λ. Since we reject the null hypothesis, the value for the leveling kernel in this case is 0. If this value is less than one, then the leveling kernel value is equal to the Epanechnikov kernel mentioned above.

After both kernels are calculated, they are multiplied together to achieve the final weight for each bin:

$$K(u) = K_1(u_{distance})K_2(u_{level})$$

An estimate of each bin's land value is calculated using a weighted average of the bin's neighbors that have a similar land value. This is the Nadaraya-Watson estimator:

$$\hat{\theta}(\text{lat}, \text{long}) = \frac{\sum_{i=1}^{n} K(u_i)\theta_i}{\sum_{i=1}^{n} K(u_i)}$$

We iteratively repeat this process for increasing bandwidths in the distance kernel using our estimators from the previous round as input. We start with small bandwidths so we are first only considering nearby bins in our weighted average. Then, we gradually increase the number of bins considered in each iteration by increasing the distance bandwidth.

More details on this method are available in Polzehl & Spokoiny's "Propagation-Separation Approach for Local Likelihood Estimation."

Here is pseudocode for the adaptive weights smoothing algorithm. Full code is available in the project Jupyter notebook.

```
def get_kernel_estimator():
  kernels = average sale price of vacant land in each geographic bin

    for bandwidth in bandwidths:
        for bin i in bins:
            for bin j in bins:
             K1_ij = K(Manhattan distance between bin i and bin j)
             K2_ij = K(Level statistic between bin i and bin j)
             K_ij = K1*K2
          theta_i = Weighted average of all other j using K_ij as weights
    return theta_1 through theta_n
```

3.3 Log-Linear Sale Price Regression and Two-Part Regression

In the initial phases of this project, I recreated the City of Philadelphia's log-linear regression methodology by supplementing their 2023 tax year data with kernel-estimated land values.

The City of Philadelphia splits their properties into six building categories (residential, commercial, vacant, etc.) and 565 geographic subwards. They then build a model for each building type in each subward using log-linear regression on the available attributes for each property. Since there are six property types, this results in 3390 unique models.

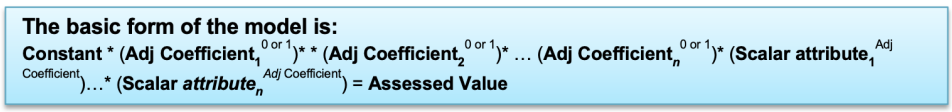

Figure 1: City of Philadelphia property tax assessment methodology

This modeling strategy can be represented mathematically as follows:

$$
\begin{aligned}
\log y &= \beta_0 + \beta_1 \log x_1 + \beta_2 \log x_2 + ... + \beta_i x_i + \beta_{i+1} x_{i+1} + ... + \epsilon \\
y &= \exp(\beta_0 + \beta_1 \log x_1 + \beta_2 \log x_2 + ... + \beta_i x_i + \beta_{i+1} x_{i+1} + ... + \epsilon) \\
y &= \exp(\beta_0 + \log x_1^{\beta_1} + \log x_2^{\beta_2} + ... + \beta_i x_i + \beta_{i+1} x_{i+1} + ... + \epsilon) \\
y &= \beta_0' * x_1^{\beta_1} * x_2^{\beta_2} * ... * \beta_i'^{x_i} * \beta_{i+1}'^{x_{i+1}} * ...
\end{aligned}
$$

Where x_1 through x_{i-1} are continuous variables and x_i through x_m are indicators for categorical variables.

Rather than creating individual models for each subward, I included the kernel estimated land value and created one model for the entire City of Philadelphia for

residential units. After excluding outliers, this model performed at industry standards for median ratio of assessed to actual sale price and coefficient of dispersion. The model was outside of industry standards for price-relate differential with a value of 1.05 (goal is of between .98 and 1.03). (See the Section 3.5 for more details on these metrics)

Additionally, I created a two-step regression model where the kernel-estimated log land price per foot was subtracted from the actual log sale price per foot, and the residuals were regressed on using the features of each example.

$$\log y^i - \log \theta(\text{lat}^i, \text{lon}^i) = \beta_0 + \beta_1 x^i + \epsilon$$

The goal of this model is to provide a separate estimate of land value and building value.

After predicting the difference in sale price and land value, the land value was added back in to get the final prediction of sale price. This final value was used to calculate the ratio of assessed to actual property value.

3.4 Cross-Validation and Hyperparameter Tuning

3.4.1 Adaptive Weights Smoothing

The adaptive weights smoothing algorithm has two important parameters: the list of distance bandwidths to iterate through (starting from small values to larger values) and the leveling bandwidth. Additionally, the size of the geographic bins impacts how many observations of vacant land sales occur in each bin - smaller bins results in more outliers with either very high or very low vacant land sale price relative to their neighbors.

The list of bandwidths used were decided by a combination of cross-validation and visual inspection of the estimated land value maps. As mentioned above, the assumption of the adaptive weights smoothing algorithm is that land values are similar between adjacent bins and that land values increase smoothly. If bandwidths for the distance kernel or the leveling kernel are too small, then not enough bins will be considered when calculating a weighted average. This causes discontinuities where several bins have very different estimated land values than their neighbors. This can be striking since lower valued bins will "pull each other down" based upon the leveling kernel, and can result in obviously incorrect pockets of very low-valued land:

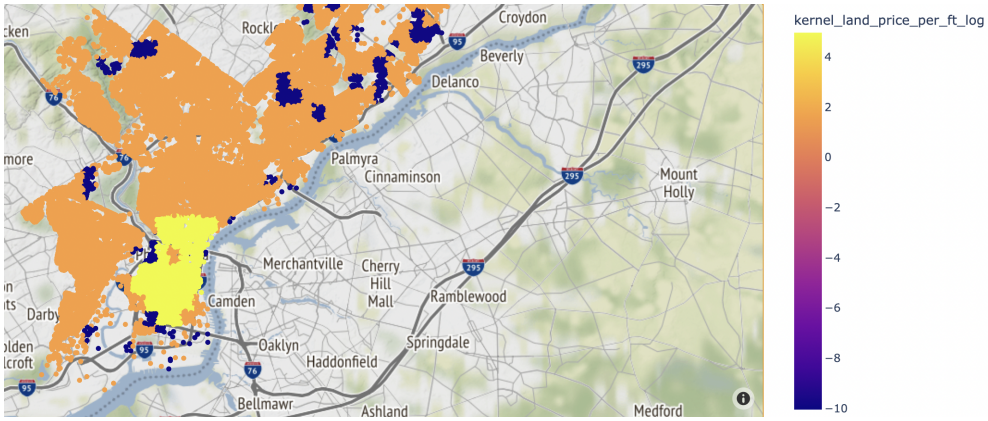

Figure 2: This is an example of poor hyperparameter choices, resulting in several discontinuities as well as pockets of low-value land in downtown

Better hyperparameter choices result in more consistent land values with fewer discontinuities.

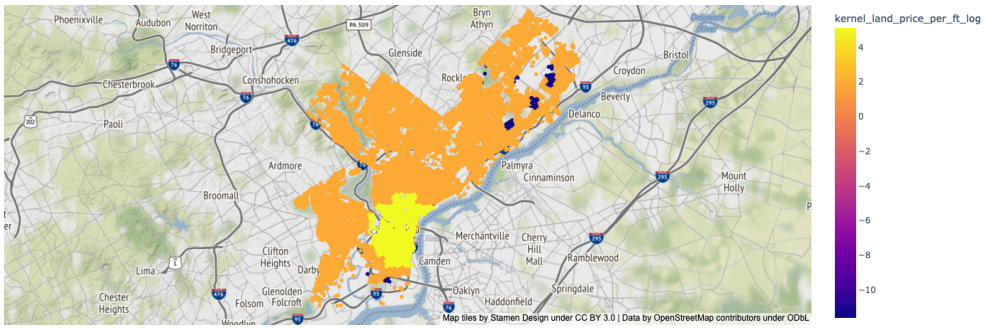

Figure 3: This is an example of better hyperparameter choices, resulting in fewer discontinuities

As a sanity check, we can compare the estimated land values to the actual vacant lot sale prices. Each of these is represented as the log of sale price per square foot. For the actual values, we have a maximum value of 8.34, a minimum value of -14.86, a mean value of 1.83, and a median value of 1.00. For the estimated values, we have a maximum value of 5.45, a minimum value of -13.26, a mean value of 1.66, and a median value of 2.54.

Based upon this, we can see that the kernel estimation smooths off the maximum and the minimum values, and that it brings up the median quite a bit. This is not surprising since there are quite a few pockets of very cheap vacant land transactions. The kernel mechanism will "bring these values up" if they are near other pockets of cheap - but not quite *as cheap* - vacant land.

3.4.2 Regression Models

I tested multiple different regression models to predict the log of the sale price per foot. I used Sci-Kit's ordinary least squares, Ridge, and Lasso models. I also used XGBoost. XGBoost is a decision-tree ensemble model that iteratively adds decision-trees that learn the residuals from the previous step — each new tree focuses on the errors that the previous ensemble model made.

I used standard 5-fold cross-validation and evaluated several metrics to tune models. For Ridge and Lasso, I used a regularization constant of .1.

To speed up hyperparameter tuning with XGBoost, I used a fraction of the test set. The full test set contained about 500,000 examples, and I used 100,000 examples for initial hyperparameter tuning. Results during cross-validation stabilized between 50,000 an 100,000 examples. The final hyperparameters I used for XGBoost were a learning rate of .1, a maximum tree depth of 8, and 180 total estimators.

I also used XGBoost's feature importance score to select which features to include in the final model. I used cross-validation, and determined that including about half of the available features gave the best results.

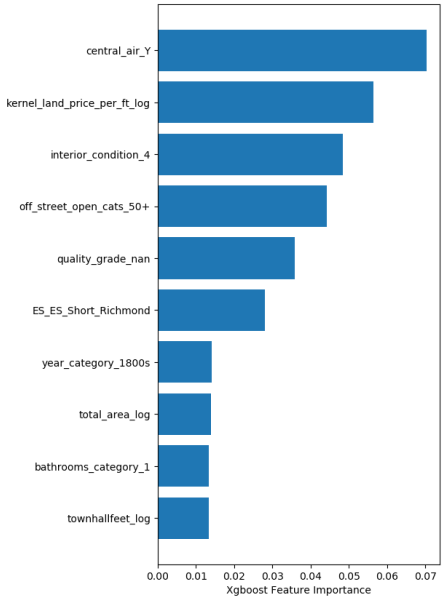

Figure 4: The top ten features ranked by XGBoost's feature importance function on the training data. Note that the kernel land price feature is ranked second.

3.5 Scoring

Each of these models uses mean squared error as its primary loss function, but I also evaluated the models several real estate specific statistics. Additionally, the mean absolute error is particularly salient in estimating real estate values. People have an intuition that being off by a few tens of thousands of dollars when estimating a real estate transaction is probably "pretty good."

The City of Philadelphia uses three main metrics to evaluate its assessments:

- **Ratio of assessed value to sale price**: Industry standard is for the median ratio to be between 90%-110% (a 10% margin of error). Philadelphia targets a range of 95%-102%.

- **Coefficient of Dispersion**: The coefficient of dispersion is a real estate specific measure of the variation in assessed price to actual price. The goal value is less than 15% for jurisdictions like Philadelphia. This can be thought of as "the average amount by which the ratio of assessed to actual sale value differs from the median ratio of assessed to actual sale value." The coefficient of dispersion can be represented as follows, with $\hat{yy}$ representing the ratio of assessed to actual sale price and n representing the number of observations:

$$\text{COD} = \frac{100}{n} \frac{\sum_{i=1}^{n} \hat{yy} - median(\hat{yy})}{median(\hat{yy})}$$

- **Price-Related Differential**: The price related differential is a real estate specific measure of how lower and higher value properties are assessed.

  The numerator is an unweighted average of all of the ratios of assessed to actual sale price. The denominator is a weighted average of all of the ratios of assessed to actual sale price with the weights corresponding to the value of the property.

If higher valued properties have an average "assessed to actual" ratio less than lower valued properties, then the weighted average will be less than the unweighted average. This will cause the price-related differential to be greater than one since the denominator will be less than the numerator. If lower-valued properties have an average "assessed to actual" ratio less than higher-valued properties, then the price-related differential will be less than one.

Values between .98 and 1.03 are considered industry standard.

3.6 Model Selection

For my final models, I used an ordinary least squares model and XGBoost. The ordinary least squares model provided results comparable to Ridge and superior to Lasso, and its results are highly interpretable. Since interpretability is particularly important in property tax assessment, I chose to use ordinary least squares rather than one of the regularized regression models. XGBoost was the best-performing model, so I chose to use this as a performance benchmark.

I also ran four different versions of each model. The first includes the kernel estimation for land value as a feature. The second excludes the kernel estimation for land value as a feature. This second version, however, includes other proxies for land location like elementary school district, distance to townhall, number of street parking spaces, etc.

The third version subtracts the kernel estimation for land value, regresses on the residuals, then adds the value back in. The idea here is that the building features can be used to predict the building value and the land value is predicted separately. See section 3.3 for more details.

The fourth version does the same two-step regression as the third, but it excludes other features that are proxies for location (like elementary school district).

4 Results & Analysis

		Median Ratio	Mean Ratio	COD	PRD	MAE	R2
INCLUDE KERNEL	OLS	0.95	1.08	13.37	1.05	45,471	0.77
	XGBoost	0.95	1.05	11.43	1.05	39,542	
EXCLUDE KERNEL	OLS	0.95	1.08	13.33	1.05	45,346	0.77
	XGBoost	0.94	1.05	11.76	1.05	40,161	
RESIDUALS	OLS	1.04	1.21	15.88	1.19	67,252	0.67
	XGBoost	0.95	1.08	12.98	1.14	45,015	
RESIDUALS (FEWER COLS)	OLS	1.00	1.25	24.76	1.26	91,594	0.45
	XGBoost	0.99	1.17	18.29	1.23	66,767	

Figure 5: Results on testing data of 56,150 examples. "Ratio" refers to the ratio of assessed sale price to actual sale price. The top and bottom 10% of assessed to actual sale price were dropped as outliers. More detailed explanations of the different regression tasks in section 3.3

When evaluating the results, I dropped the top and bottom ten percent as outliers based the ratio of assessed price to sale price. In the City of Philadelphia's assessment guidelines, they mention that they manually adjust outliers, although they did not give information on what percentage of their data they consider to be outliers.

The best performing model was XGBoost that included estimated land values as a feature. After dropping outliers, this model met industry standards for median ratio of assessed to actual sale price as well as coefficient of dispersion. It missed industry standards for price related differential (target is .98-1.03). If the price-related differential is greater than one, then higher-valued properties are over-assessed relative to lower-valued properties. So, this model over-assessed higher valued properties. Additionally, the mean absolute error was $39,542.

For the two-step regression, XGBoost is also able to achieve results comparable to ordinary least squares on more standard log-linear regression. However, XGBoost is not an easily interpretable model, so the likelihood of adoption is low. For two-step regression using ordinary least squares, I achieved a median assessed to sale price ratio of 1.04, which meets industry standards. However, the coefficient of dispersion (15.88 - target is less than 15) and price-related differential (1.19 - target is between .98 and 1.03) did not meet standards.

For two-step regression excluding all location-based features (like elementary school, distance to townhall, etc.) ordinary least squares is able to achieve a median actual to assessed ratio of 1.0, but its coefficient of dispersion (24.75%) and price-related differential (1.26) are well outside of industry standards.

Using ordinary least squares, we can calculate what percentage of the assessed price is attributable to the estimated land value. Using the weights associated with kernel land value, the mean and the median ratio of weighted kernel land value to predicted sale price are both .02, indicating that the model is allocating about 2% of its prediction to the kernel land value. These numbers are quite low - likely because we include many other features that are proxies for location and land value like elementary school district and distance to townhall.

If we compare the estimated land value to the predicted sale price without using weighting, we get a mean value of 31% and a median value of 18%. The City of Philadelphia uses a blanket estimate of 20% for land value, so this indicates that the City is likely underestimating the percentage of property value attributable to land.

5 Future Work

The most obvious next step is to try to improve the two-step models to reach industry standards. Given that the adaptive weights smoothing algorithm generates three or four price per square foot "levels" of land value on the City of Philadelphia data, a possible first step is to generate three or four different two-step models — one for each "level" of land.

Since the models presented here are close to industry standards, further feature engineering and hyperparameter tuning could plausibly yield results that are usable by property tax assessors.

Additionally, black box methods like neural networks could be used to perform the regression tasks in this work. The lack of interpretability of neural network results presents a problem for widespread adoption. However, good results from neural networks could be a benchmark to shoot for with other modeling techniques. Additionally, neural networks could be used to generate synthetic data to fit other more interpretable models.

6 Discussion of Effort

This project involved three major phases of effort: cleaning the dataset from the City of Philadelphia and understanding their methodology, implementing the adaptive weights smoothing algorithm, and tuning models and implementing scoring functions.

The most difficult parts of the project were piecing together the methodologies for the City of Philadelphia's property tax assessments and the kernel regression used in the Kolbe paper.

While Philadelphia has quite a bit of data freely accessible, the documentation for what is actually going on in the data is scattered around their website. Also, as with most large data sets like this, there are a lot of inconsistencies (mixed data types, inconsistent handling of missing values, et.c) and issues with the data (things like latitude and longitude being flipped, typos, etc.) that required cleaning. Additionally, the methodology that they used for their market value assessment is not explained clearly and required some sleuthing to figure out.

Property tax assessment also uses domain specific statistics like the coefficient of dispersion and the price-related differential that were not explained in the documentation on the Philadelphia website.

Similarly, the methodology for the adaptive weights smoothing was not very clear. While most of the information is in the paper, it was scattered throughout so it required several re-readings and re-implementations to understand and implement the methodology. I also read papers specifically on the adaptive weights smoothing model and statistical theory behind the kernels used in the adaptive weights smoothing protocol.

The models actually reasonably quickly, so I was able to try several different models and combinations of parameters during the cross-validation process.

In terms of skills and prior experience, I was somewhat familiar with the regression techniques used as well as XGBoost. The adaptive weights smoothing algorithm was new.

6.1 Work Done (But Not Included)

I tried two additional regression methods that I did not include in the final output. I coded a Ridge regression model that could selectively exclude certain features from regularization. This model used gradient descent to converge to a solution. The results for this model were much worse than for the other linear regression models and its behavior was somewhat unstable, so I chose to exclude it. I also used a TabTransformer model to attempt regression. While I wanted to include a deep learning model as one of my regressors, I was not able to get this working properly and decided it was outside of the scope of this project.

7 References

1. Chen, T. and Guestrin, C.: 2016, XGBoost: A Scalable Tree Boosting System, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794

2. City of Philadelphia, Office of Property Assessment Mass Appraisal Valuation Methodology Overview Tax Year 2023

3. Colwell, P. F. and Munneke, H. J.: 2003, Estimating a price surface for vacant land in an urban area, Land Economics 79, 15–28.

4. Epanechnikov, V. A.: 1969, Non-parametric estimation of a multivariate probability density, Theory of Probability & Its Applications 14, 153–158.

5. Helbing, G., Shen, Z., Odening, M. and Ritter, M.: 2017, Estimating location values of agricultural land, The German Journal of Agricultural Economics 66, 188–201.

6. Kolbe, J., Schulz, R., Wersing, M. and Werwatz, A.: 2015, Identifying Berlin's land value map using adaptive weights smoothing, Computational Statistics 30, 767–790.

7. Kolbe, J., Schulz, R., Wersing, M. and Werwatz, A.: 2019, Land value appraisal using statistical methods, Agricultural Land Markets - Efficiency and Regulation, 11-26.

8. Nadaraya, E. A.: 1964, On estimating regression, Theory of Probability & Its Applications 9, 141–142.

9. Open Data Philly: Philadelphia Properties and Assessment History dataset: https://www.opendataphilly.org/dataset/opa-property-assessments

10. Peterson, R. and Carter, T.: 2010, Measuring Real Property Appraisal Performance In Washington's Property Tax System 2009, 27-28.

11. Polzehl, J. and Spokoiny, V.: 2006, Propagation-separation approach for local likelihood estimation, Probability Theory and Related Fields, 135, 335–362.

12. Yalpir, S. and Unel, F.B.: 2017, Use of Spatial Analysis Methods in Land Appraisal; Konya Example, 5th International Symposium on Innovative Technologies in Engineering and Science 29-30 September 2017, 1575-1580.