Estimating Costal Property Values in Florida

Langyi Tian

1 / 32

Executive summary

Objective

- Organizational mission: Quantify the financial risk of tidal flooding to address stakeholder concern
- Task for this research: Make the numbers more accurate with market value estimation

Data

- Administrative property records (3 million), transaction records from ATTOM Data Solutions
- Demographic data from census data
- Flooding projections from National Oceanic and Atmospheric Administration (NOAA)

Methodology

- Build seperate models within each city and county
- Regularization models (Ridge, LASSO) as baseline
- Regression trees e.g. random forest and gradient boosting as comparison for trial models

Findings

- Trees work better than baseline
- Random forest algorithm consistently outperformed by gradient boosting
- City-level models have varying performance
- Ability to predict within 10% deviation in some cities

Project roadmap

1. Data preparation
2. Exploratory analysis
3. Seperate modeling: one model for each city/county data
4. Feature selection with individual models
5. Build seperate predictive models for 85 city-level subsamples
6. Functionalities to test and parse out performance metrics for regularized models and regression trees
7. Personalize data filtering parameters
8. Iteration through cities/counties and view corss-validated model performance, map county-level model performance

1. Data preparation

Select features from property records in a real estate broker's database.

```
#Select features to import, subset and save
home_dta <- select(
  home_dta_original,
  attomid,#Matching ID
  deedlastsaleprice,#Transaction price last sale
  situsstatecode,#State code
  situscounty,#County code
  propertyaddresscity,#City code
  ownertypedescription1,#First owner is individual/company?
  ownertypedescription2,#Second owner is individual/company?
  deedlastsaledate,#Date of market sale
  yearbuilt,#Year when built
  propertyusegroup,#Commercial/residential?
  areabuilding,#Living area in sq.feet
  censustract,#Census tract division
  propertylatitude,#Lat of property
  propertylongitude,#Lon of property
  roomsatticflag,#See below
  parkinggarage:communityrecroomflag#A series variable measuring physical attributes of the property, including room
)
```

1. Data preparation

Select features from environmental risk and demographic data set constructed by Porter

```
risk_dta <-
  risk_dta_original %>% select(attomid = ATTOM_ID, #ID
                               dist_coast, #Distance to coast
                               mdkt32, #Flooding probability estimate in next years
                               Totpopbg:near_reading_rates, )#a set of demographic information varying by census tra
```

Get a simple feature of transaction frequence from transaction records

```
trans_dta <-
  trans_dta_original %>% group_by(attomid) %>% summarize(trans_times = n())#Number of transactions
```

```
## Up to here, the data dimension is 3327923 * 203
```

1. Data preparation

- Drop all unary features

```
## Up to here, the data dimension is 3327923 * 91
```

- Drop variables with too many levels, besides those are numerical

```
## [1] "hvacheatingdetail"
## [1] "exterior1code"
## [1] "roofmaterial"
## [1] "roofconstruction"

## Up to here, the data dimension is 3327923 * 87
```

- Recoded characters to factors

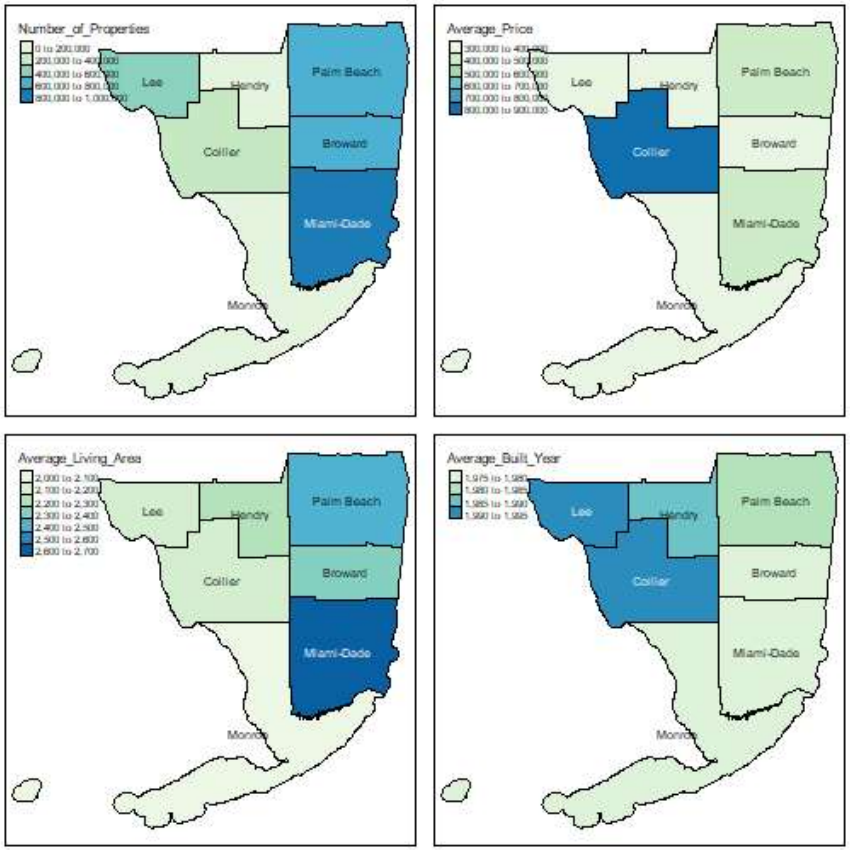
```
## These variables are recoded to factors

## [1] "situsstatecode"
## [1] "situscounty"
## [1] "propertyaddresscity"
## [1] "ownertypedescription1"
## [1] "ownertypedescription2"
## [1] "propertyusegroup"
## [1] "viewdescription"
## [1] "porchcode"
```

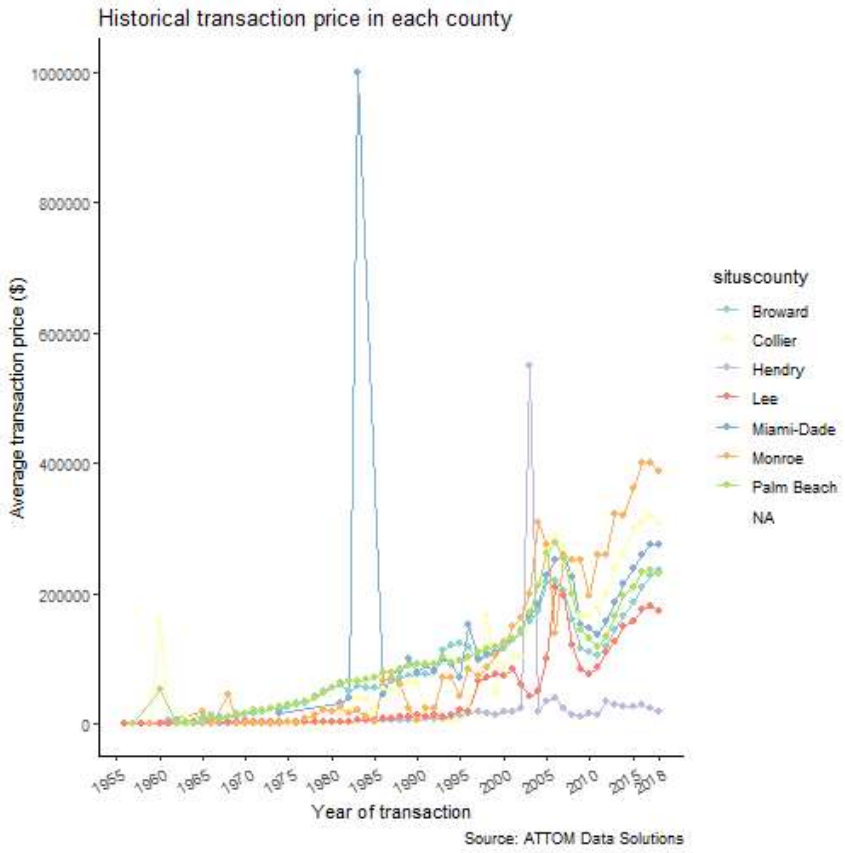
- Hold out cities with sample size too small to go into tree model.

2. Exploratory analysis

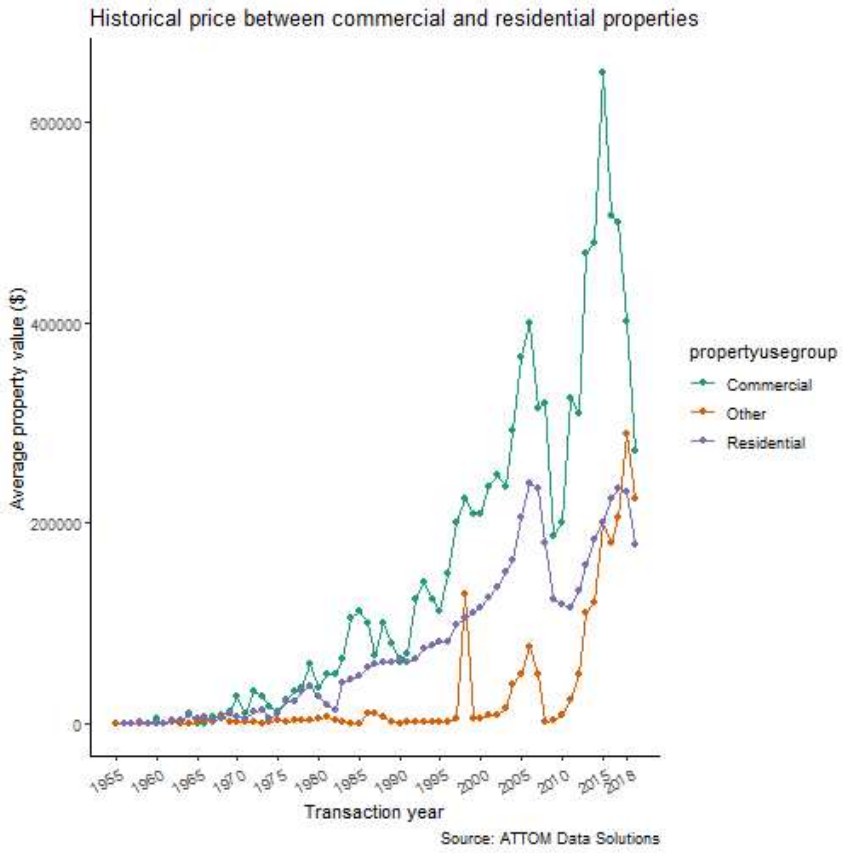
- Summarize a few key variables by county to see the geographical variation



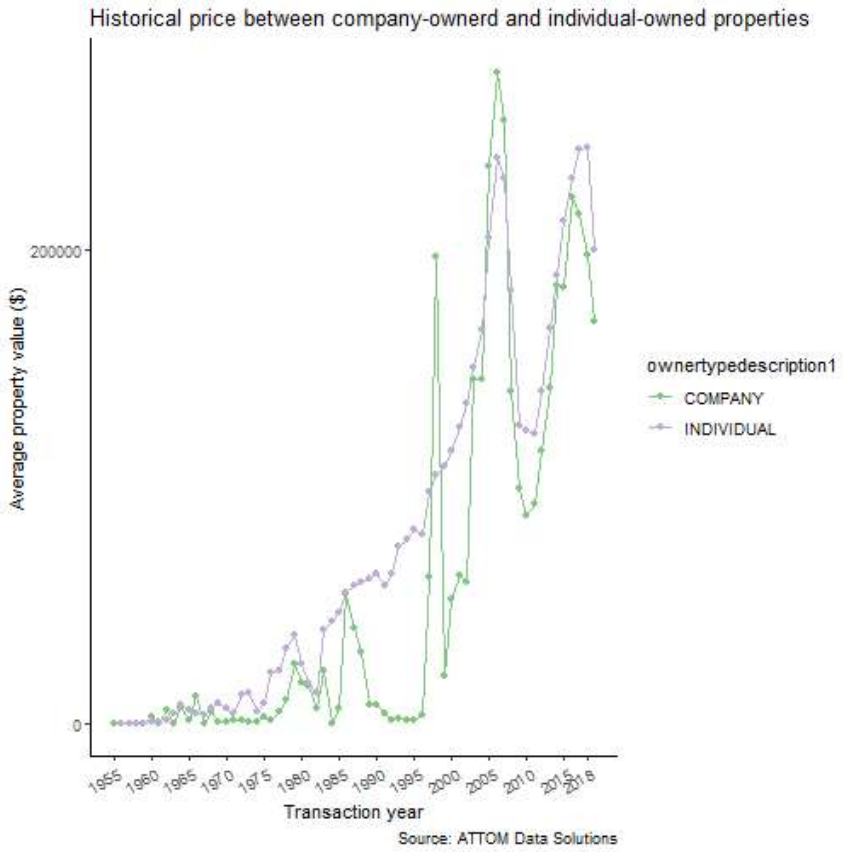
2. Exploratory analysis



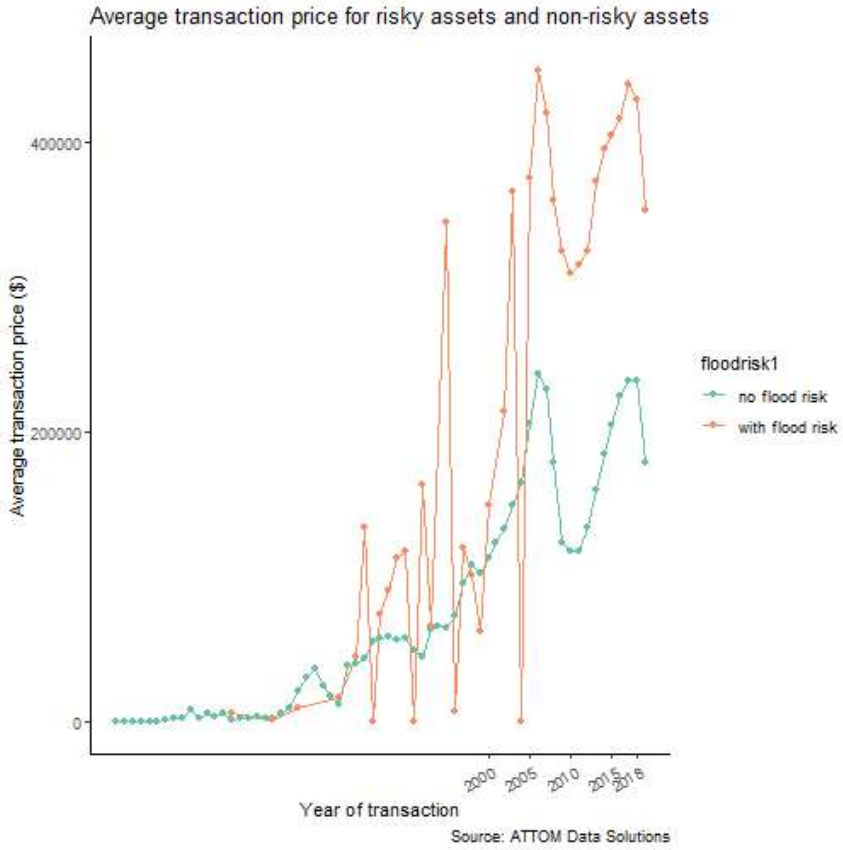
2. Exploratory analysis



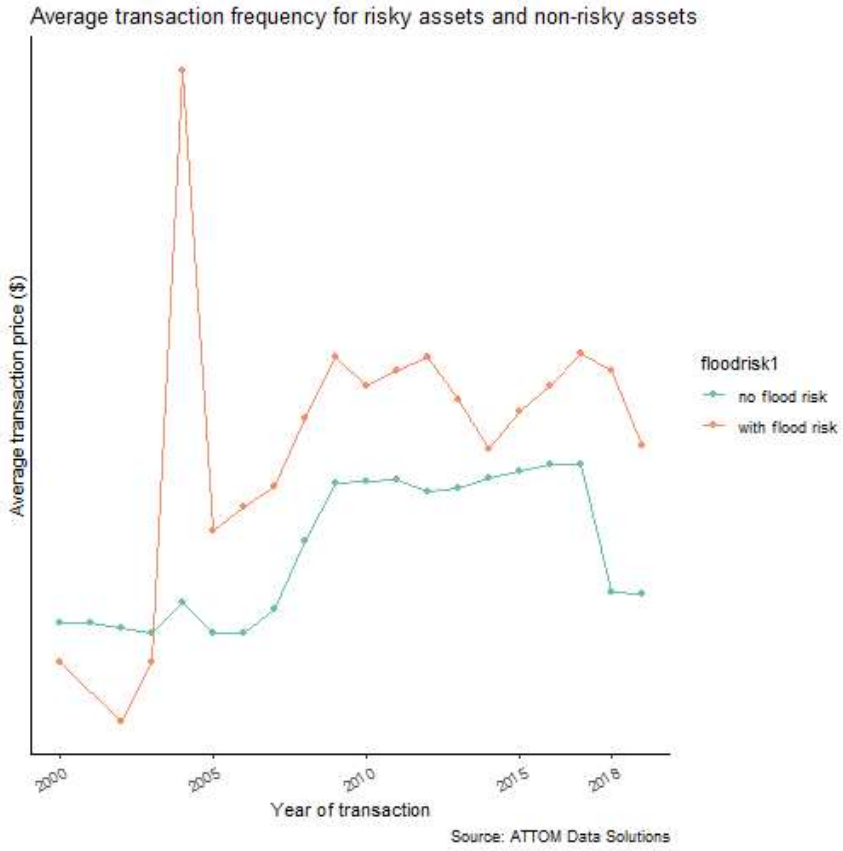
2. Exploratory analysis



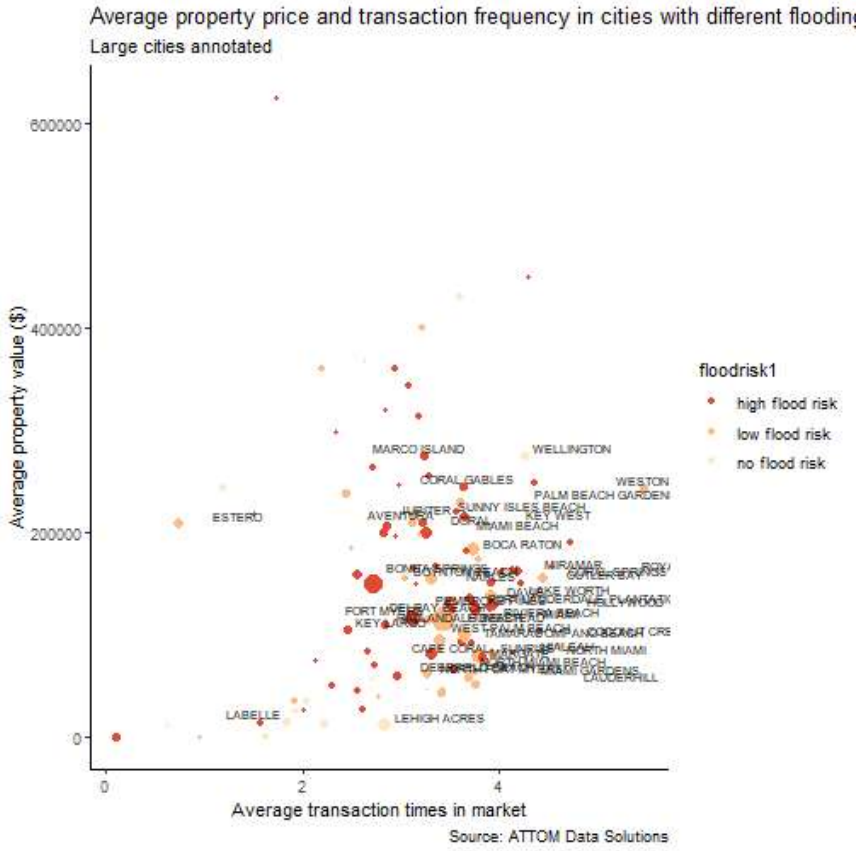
2. Exploratory analysis



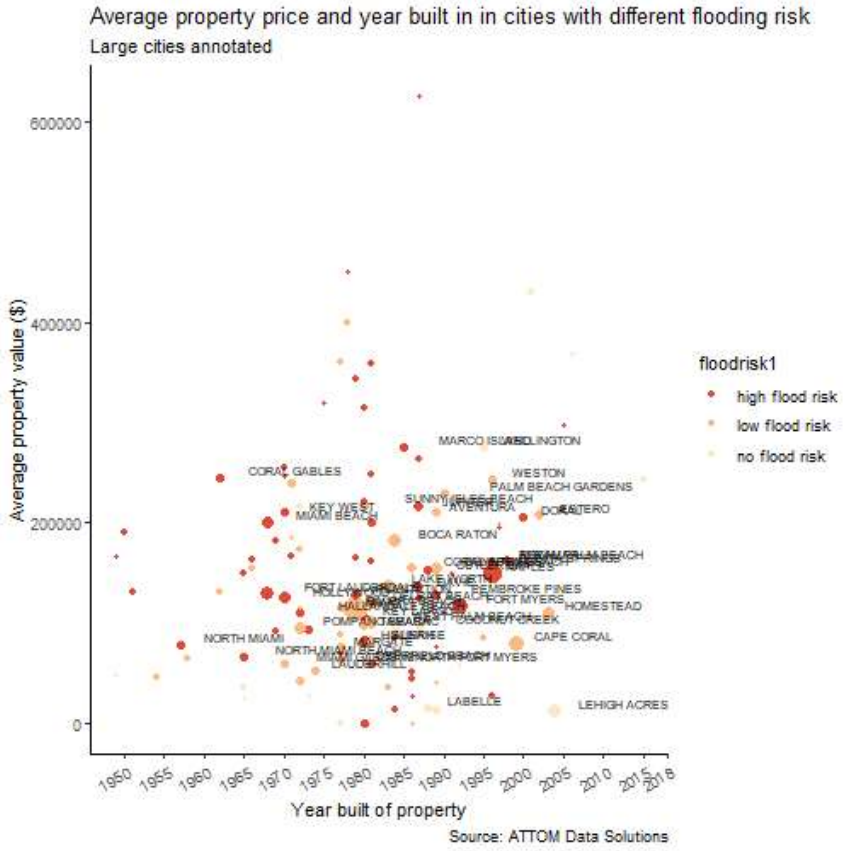
2. Exploratory analysis



2. Exploratory analysis



2. Exploratory analysis



3. Seperate modeling: one model for each city/county data

- For every city/county in South Florida, we fit an individual model to it seperately.
- Inspired by the "submarket" notion (the housing prices between neibourhoods, cities and counties vary a lot)
- Feature structure vary by county
- There are some variables that represent housing attributes that miss in data with different cases between counties.
- The process for county-level modeling is rather straightforward.

4. Feature selection with individual models

- Build function to automatically drop variables that are not important and are missing over 10% values, as we don't wish too many obs are omitted in tree model due to missing value.
- Apply the selection functions built just now to all subsamples
- Divided subsamples by county

```
summary(home_county_cleaned_dta)
```

```
##            Length Class      Mode
## Broward    79     data.frame list
## Collier    72     data.frame list
## Hendry     71     data.frame list
## Lee        79     data.frame list
## Miami-Dade 82     data.frame list
## Monroe     78     data.frame list
## Palm Beach 83     data.frame list
```

4. Feature selection with individual models

- First 5 divided subsamples by city

```
summary(home_city_cleaned_dta)%>%head()
```

```
##                   Length Class      Mode
## ALVA              70     data.frame list
## AVE MARIA         62     data.frame list
## AVENTURA          73     data.frame list
## BAL HARBOUR       68     data.frame list
## BANYAN VILLAGE    15     data.frame list
## BAY HARBOR ISLANDS 69    data.frame list
```

6. Build predictive models for individual subgroups

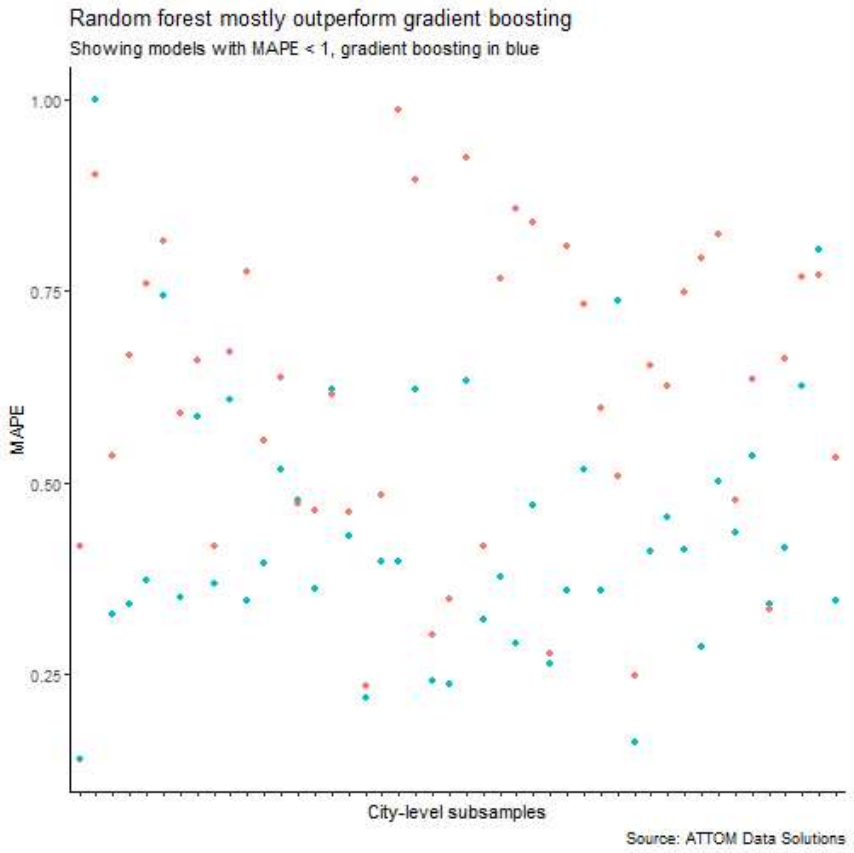
Final Ppreprocessing of individual model before modeling

- Develop functionality for final preprocessing before feeding data into model
- Here we have within each model the control of missing values and division of training/test set
- Therefore, the wragling process was automized and run in each subsample

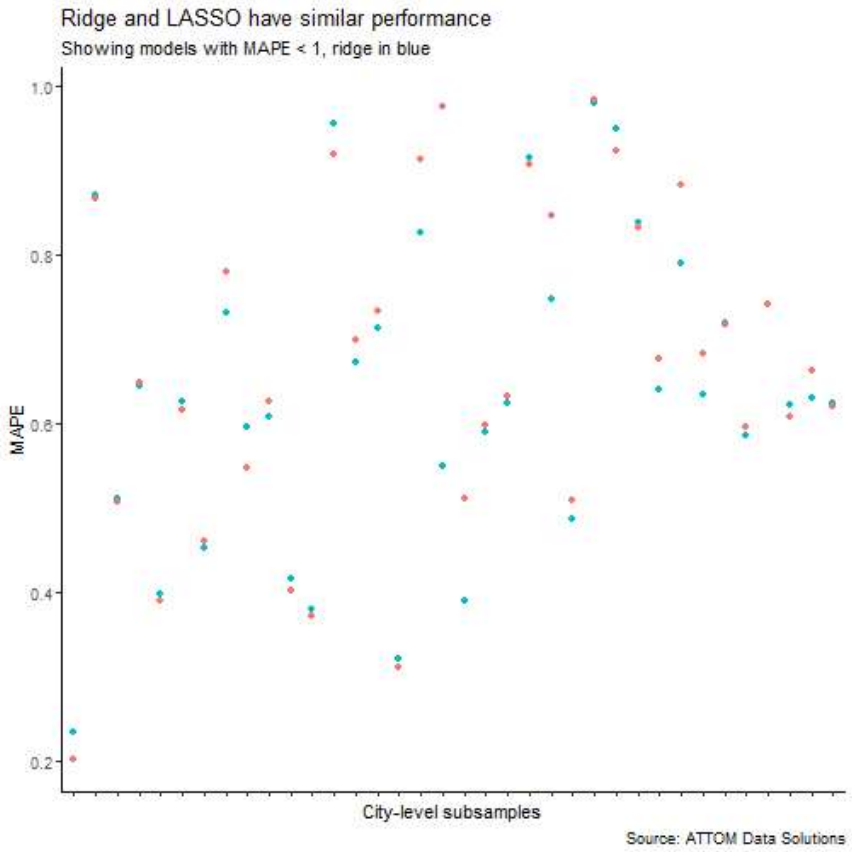
Modeling functionalities

- Regularization (Ridge and LASSO) as baseline, try also trees
- Random forest deal with data with large variations well
- Gradient boosting is good at learn specific models
- A model comparison with output from random forest and gradient boosting (also Ridge and LASSO)
- The function will parse validated key performance metrics and variables of importance of the models

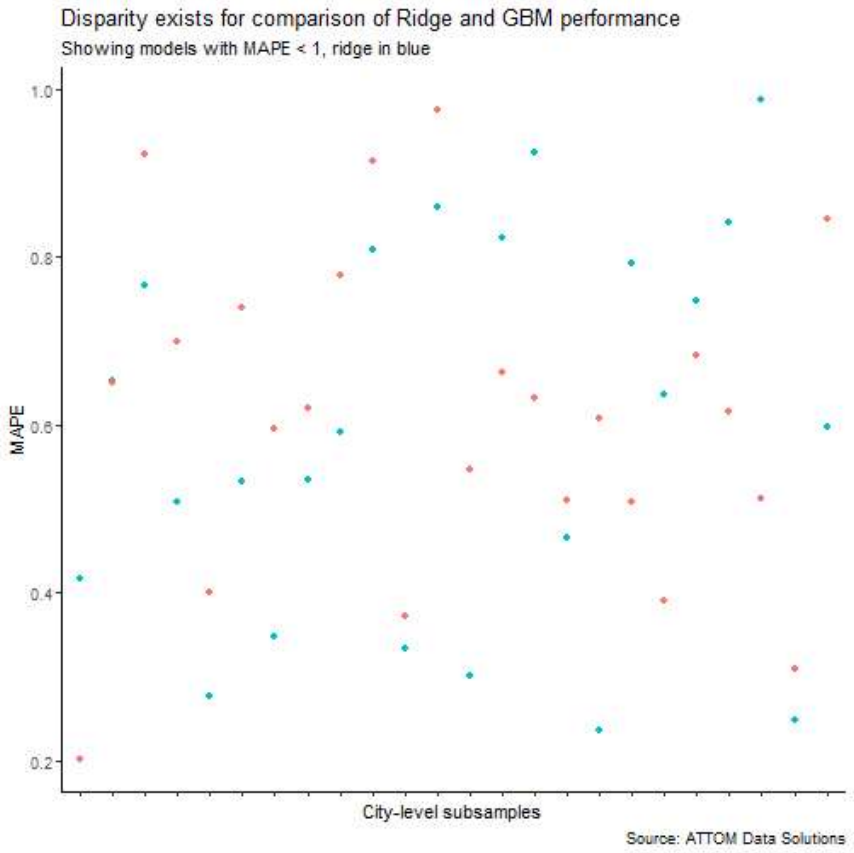
7 Visualize model performance



7 Visualize model performance



7 Visualize model performance

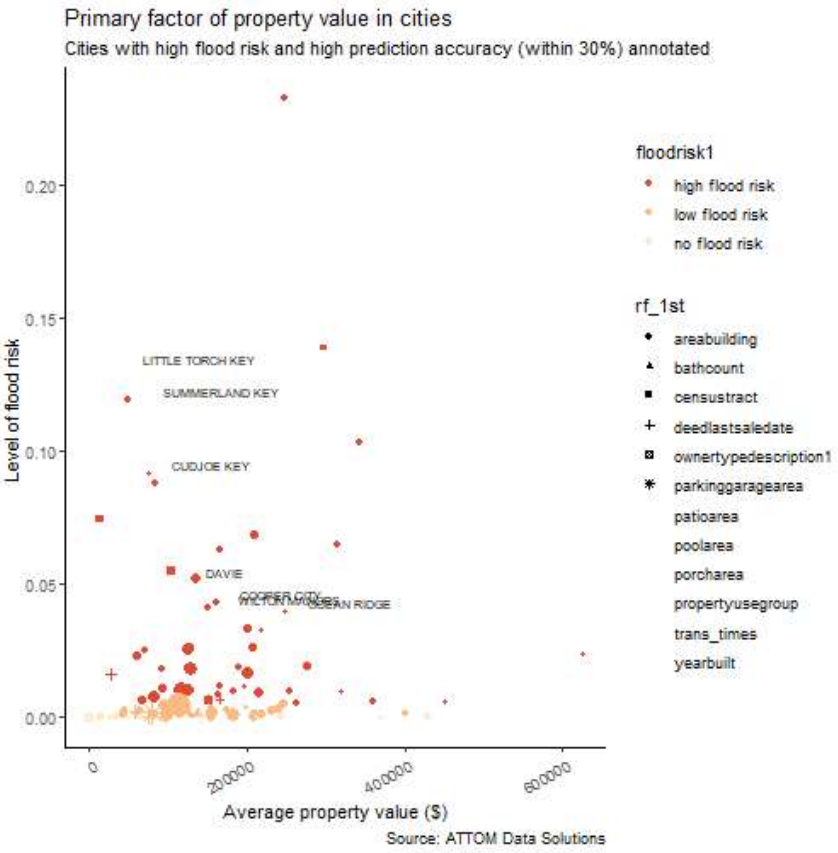


7 Visualize model performance

Show 10 entries Search:

	propertyaddresscity	num	subsample	price	area	averageprice	builtyear	
1	ALVA	8352	FL_Lee_ALVA	27000	1828	74.597892401553	1996	0.
2	AVE MARIA	2843	FL_Collier_AVE MARIA	243000	1797	152.508728179551	2015	
3	AVENTURA	19748	FL_Miami-Dade_AVENTURA	210000	1353	173.075149838635	1989	0.00
4	BAL HARBOUR	3986	FL_Miami-Dade_BAL HARBOUR	360000	1688	295.389048991354	1977	0.00
5	BANYAN VILLAGE	1556	FL_Hendry_BANYAN VILLAGE	11600				
6	BAY HARBOR ISLANDS	2825	FL_Miami-Dade_BAY HARBOR ISLANDS	155000	1266	135.947496884088	1966	0.00

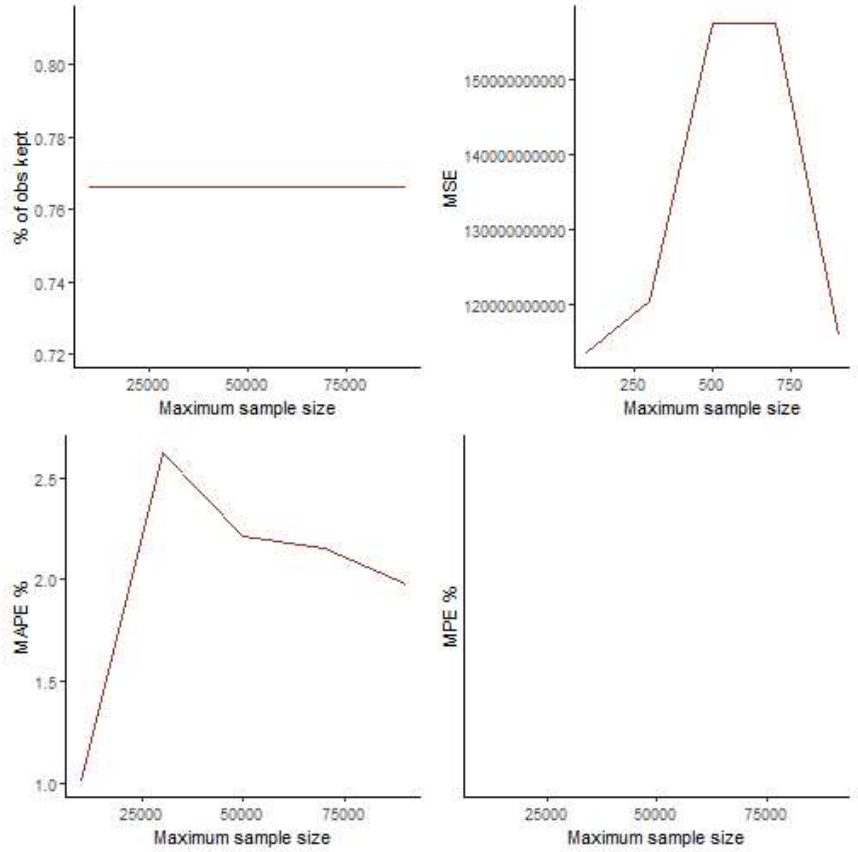
7.5 Visualize model performance



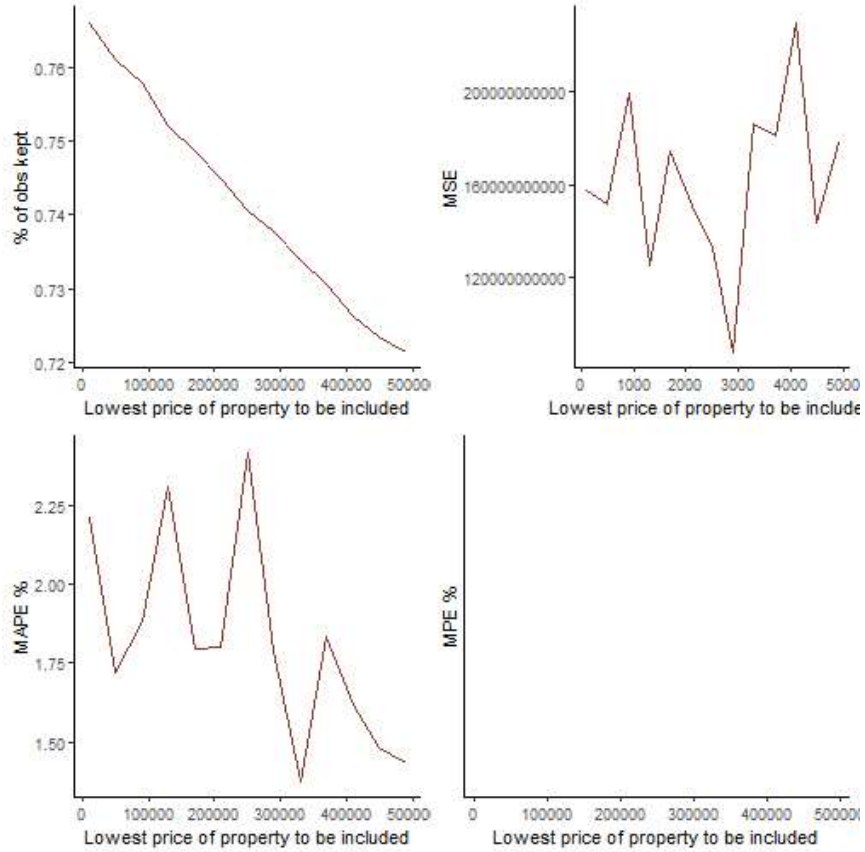
8. Personalize filtering parameters for random forest

- Filtering rule decides whether to include properties with extremely low/high price
- Cruicial to model performance
- Build functionalities to show model performance depending on filtering parameters

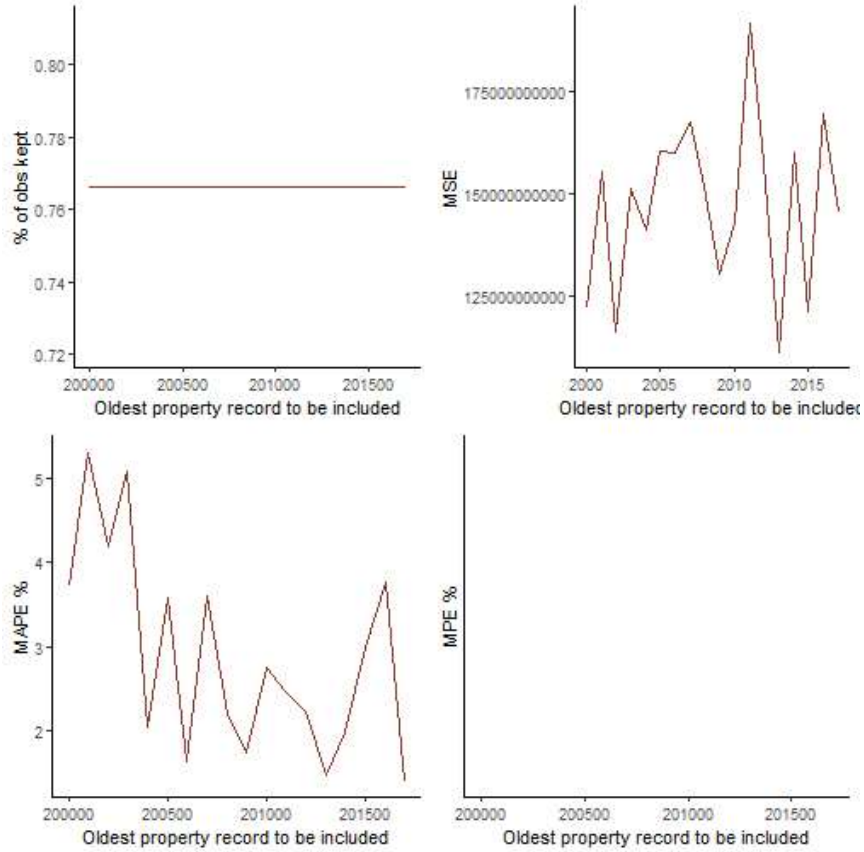
8. Personalize filtering parameters for random forest



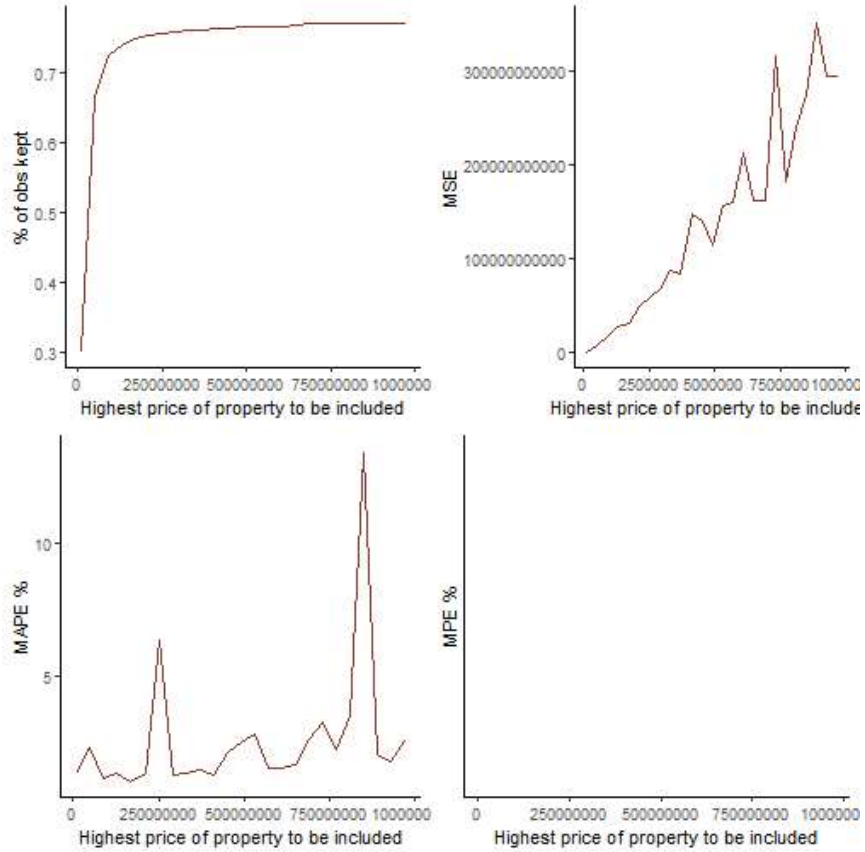
8. Personalize filtering parameters for random forest



8. Personalize filtering parameters for random forest



8. Personalize filtering parameters for random forest



9. Iteration through cities/counties and view corss-validated model performance

- Cities

Show 10 entries Search:

	price_high	price_low	year_start	size	subsample	preprocess_retention	obs
..1	5000000	100	2012	1000	FL_Collier_AVE MARIA	0.695392191347168	1000
..116	5000000	100	2012	1000	FL_Broward_WESTON	0.761011516086905	1000
..29	5000000	100	2012	1000	FL_Lee_ESTERO	0.758744028001329	1000
..96	5000000	100	2012	1000	FL_Lee_PUNTA GORDA	0.866569626394954	903
..2	5000000	100	2012	1000	FL_Miami-Dade_AVENTURA	0.764887583552765	1000
..41	5000000	100	2012	1000	FL_Palm Beach_HIGHLAND BEACH	0.89152622834085	1000
..75	5000000	100	2012	1000	FL_Broward_MIRAMAR	0.761532204038664	1000

9. Iteration through cities/counties and view corss-validated model performance

- Counties

Show 10 entries Search:

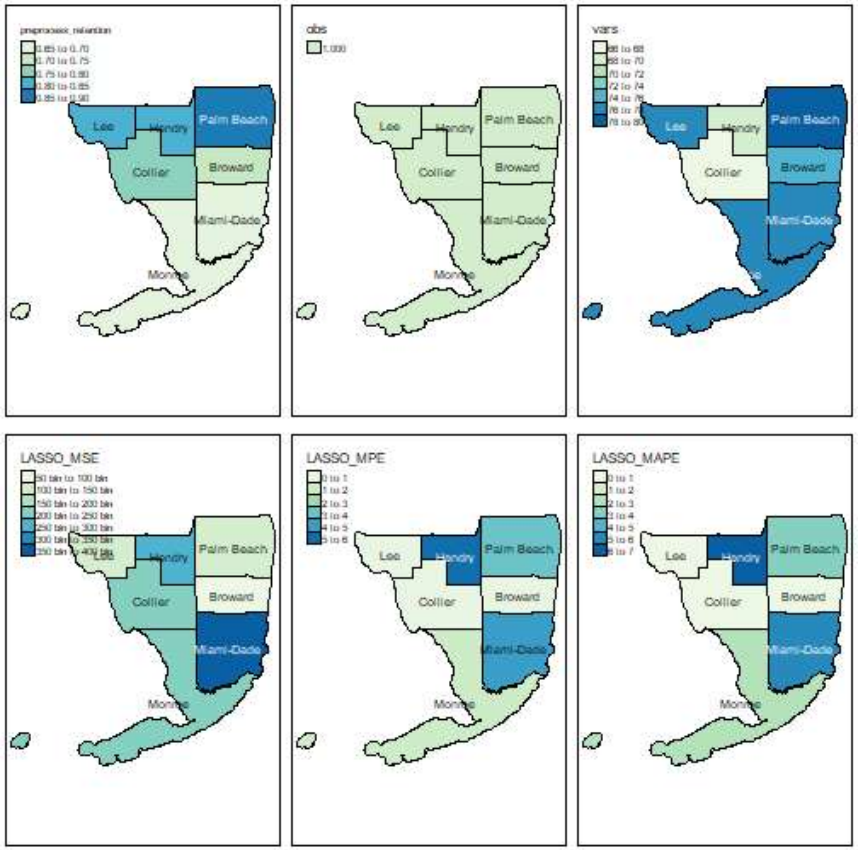
	price_high	price_low	year_start	size	subsample	preprocess_retention	obs	vars	
..1	5000000	100	2012	1000	FL Collier	0.776788380518588	1000	67	239
..3	5000000	100	2012	1000	FL Lee	0.811516660579327	1000	77	112
.	5000000	100	2012	1000	FL Broward	0.741652088787585	1000	74	602
..5	5000000	100	2012	1000	FL Monroe	0.692692644057915	1000	76	222
..6	5000000	100	2012	1000	FL Palm Beach	0.888637396105932	1000	80	104
..4	5000000	100	2012	1000	FL Miami-Dade	0.651337273099856	1000	76	376
..2	5000000	100	2012	1000	FL Hendry	0.800411273738962	1000	69	26

Showing 1 to 7 of 7 entries Previous 1 Next

30 / 32

10. Map county-level model performance

We can to see how each county performs on the map.



11. Brief Summary

- Random forest algorithm consistently outperformed by gradient boosting
- City-level models have varying performance and have potential for better model selection
- Ability to predict within 10% deviation in some cities
- Key filtering parameters optimized to improve prediction with reasonable constraint on sample size
- Transaction time and building area is 2 primary factor to determine property value

12. Next step

- Add census tract block medium price as control
- Can try tuning the filtering parameter for each model
- log scale price
- recode zip code to dummy (one-hot recoding)