Real Estate Project

Authors: Javier Vivas, Ravi Sharma, Nikat Patel, Evgeny Neashev, Kacper Krasowiak, Jordan Girard

Research topic

The size of the global professionally managed real estate market is about $8.5 trillion in 2017. Despite the advancement of the market, yet interpreting property pricing is still a challenging very difficult task.

Usually only socio-economic predictors are used to estimate the supply and demand change on the real estate market. However, the impact of online activity on real estate portals on annual price changes of real estate listings is still not well understood. In this paper, we are going to use socio-economic and listing data obtained from Realtor.com for fiscal year 2017 in order to deriveprovide an alternative-pricing model.

The research is designed to help predict the year-on-year (YoY) price change in 2017. The model would improve the understanding on the dynamic of the real estate market, which potentially could lead to meaningful business and social observations.

Research Question

This research paper is to better understand the YoY changes of residential homes prices in the United States. The potential impact of socio-economic indicators and online activity on real estate portals on YoY price changes of real estate online listings for single-family residential homes. The goal is to identify variables that could help predict the YoY price change in 2017 the most precisely.

Hypothesis

Under the economic conditions of 2017, we expect the number of affordable homes for sale to be the main driver of price growth. Our main hypothesis is higher price growth should occur in markets where inventory supply was low and dropping. Hence, inventory growth YoY and affordability should yield significant explanatory power of price. Another one of our hypotheses is growth in online searches should help tilt the balance into a sellers market and result in higher listing prices. Hence we are predicting that realtor.com page views per listing YoY would also be a significant predictor of price growth.

Data Collection

The working dataset comes from realtor.com, which is the housing research database, which includes historical aggregated monthly metrics on residential, for-sale listings proprietary to the company along with key economic indicators at market level licensed from third party providers. Listing data comes from a proprietary database that aggregates listing attributes from over 800 Multiple Listing Services (MLSs) data feeds, and produces periodic snapshots of key metrics. The initial analysis will focus on modeling annual snapshots of data for U.S. metropolitan areas in 2016 and 2017 to obtain year-over-year movement. The annual snapshots are equal-weighted averages of the monthly observations of each metric.

Methodology

In order to analyze the relationship amongst the various variables at stake, the research will implement a systematic eight-step logic for the analysis.

1. **Load Data**
   (a) Load appropriate libraries to work with Excel files
   (b) Load data in R data frames (listings data, economic data)
   (c) Verify: the dimensions, names of the variables, sample rows of data to match with source data codebook
2. **Identify relevant subset(s) of data for analysis**
   (a) Identify the appropriate subsets for further analysis
   (b) Create subset data frames in R
3. **Prepare a summary of individual variables**
   (a) Run summary statistics command on each variable of interest
   (b) Create appropriate visual(s) for summary information
   (c) Explain in brief the observations from the summary statistics and the plots
   (d) Identify any next steps due to the occured observations, suspected outliers or others
   (e) Update project documentation - important information in the main section, the other details in Appendix
4. **Part 1 - Fitting models**
   (a) Try a regression with all continuous variables (listings data)
   (b) Identify the variables that are not required, use appropriate techniques to drop those variables
   (c) Try a regression with all continuous and categorical variables (listings data)
   (d) Identify the variables that are not required, use appropriate techniques to drop those variables
   (e) Compare the final models with and without categorical variables to see which one is preferable
5. **Merge datasets**
   (a) Using the common column, merge listings and economic datasets
6. **Part 2 - Fitting models**
   (a) Try a regression with all continuous variables (economic data)
   (b) Identify the variables that are not required, use appropriate techniques to drop those variables
   (c) Try a regression with all variables from combined listings and economy datasets
   (d) Identify the variables that are not required, use appropriate techniques to drop those variables
7. **Test and compare models**
   (a) Perform diagnostic tests for all models, check for assumptions
   (b) Look for influential points
   (c) Apply fixes, where required or update/discard models
   (d) Apply bootstrapping where necessary to find out appropriate confidence intervals
   (e) Use the leading models on the next years data to test real world predictive accuracy of the model
8. **Conclusion**
   (a) Share the findings and comments

Load Data

Our study initiated with two datasets: a) listings data (called listing); and b) economic data (called economy). We will present below the name and nature of each of the variables and explain the meaning of it as needed throughout the analysis.

```
library(xlsx)
listing <- read.xlsx("HES_TEMPLATES_v2.xlsx", 1)
economy <- read.xlsx("HES_TEMPLATES_v2.xlsx", sheetIndex = 2, startRow = 1, endRow = 382)
```

Identify relevant subset(s) of data for analysis

To consider a subset comprising the top 100 ranking metropolitan areas, a smaller dataset is created, as follows:

```
listing100 <- data.frame(listing[1:100,])
dim(listing100)
```

```
## [1] 100  13
```

We now have 100 rows and 13 vaeriables in our new dataset called listing100. The reason for creating a smaller dataset is [FILL IN].

The variables for each of the datasets are:

```
names(listing)
```

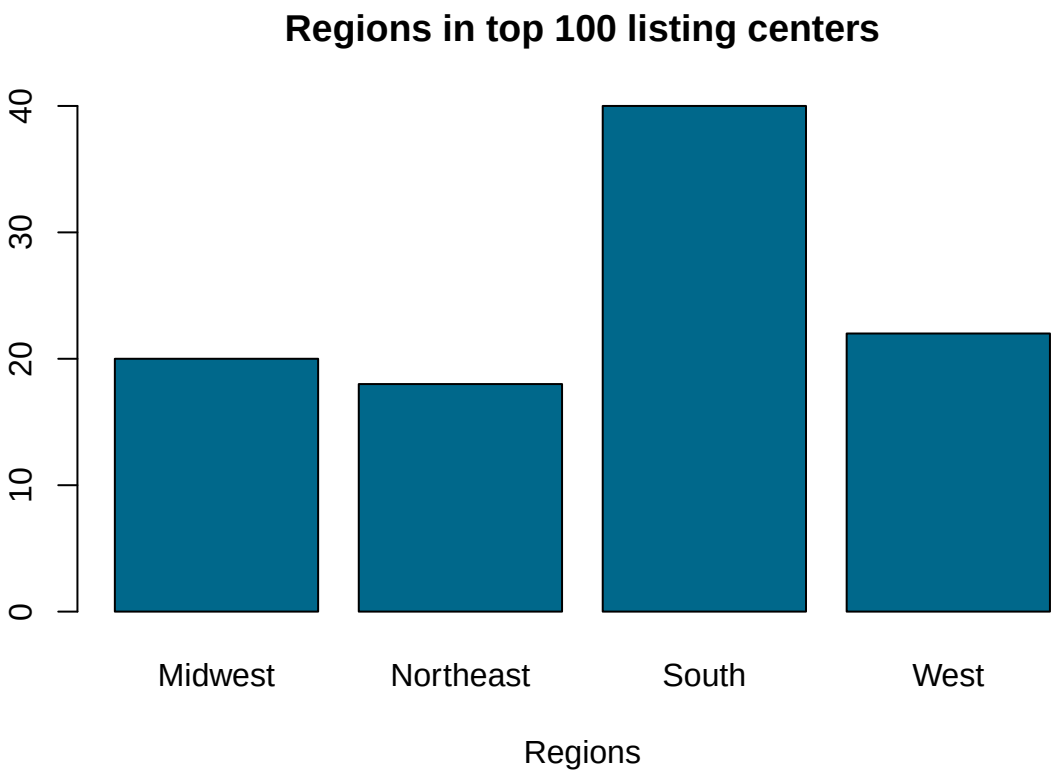
```
##  [1] "RankHH"                        "Cbsacode"
##  [3] "Cbsatitle"                     "Region"
##  [5] "Avg..Median.Listing.Price.Yy"  "Avg..Active.Listing.Count.Yy"
##  [7] "Avg..Ldpviews.Per.Property.Yy" "Avg..Median.Dom.Yy"
##  [9] "Avg..Median.Listing.Price"     "Avg..Active.Listing.Count"
## [11] "Avg..Ldpviews.Per.Property"    "Avg..Median.Dom"
## [13] "Avg.ActiveListingsper1000HH"
```

```
names(economy)
```

```
##  [1] "RankHH"                       "Cbsa.Code"
##  [3] "Cbsa.Title"                   "Region"
##  [5] "End.Of.2016.Household"        "End.Of.2017.Household"
##  [7] "End.Of.2016.Job"              "End.Of.2017.Job"
##  [9] "Income.2017"                  "Income.2016"
## [11] "Unemployment.Rate.2017"       "Unemployment.Rate.2016"
## [13] "Buy.Pct.2017"                 "Buy.Pct.2016"
## [15] "HH_yoy"                       "JOB_yoy"
## [17] "INC_yoy"                      "UNEMP_yoy"
## [19] "BUYPCT_yoy"                   "Home.Ownership.2017"
## [21] "OwnOccHH2017"                 "Sale.Px.Recovery"
## [23] "Sale.Price.Yoy"               "Sale.Px.Recovery..Msa1."
## [25] "Sale.Price.Yoy..Msa1."        "Total.Starts"
## [27] "Total.Starts.Recovery"        "Total.Starts.Yoy"
## [29] "Total.Starts..Msa1."          "Total.Starts.Recovery..Msa1."
## [31] "Total.Starts.Yoy..Msa1."
```

There are 4 regions in our dataset and they are allocated as follows:

```
barplot(table(listing100$Region), main = "Regions in top 100 listing centers", xlab="Regions",col="deeps
```



Prepare a summary of individual variables

We are going to explore each of the variable in more depth:

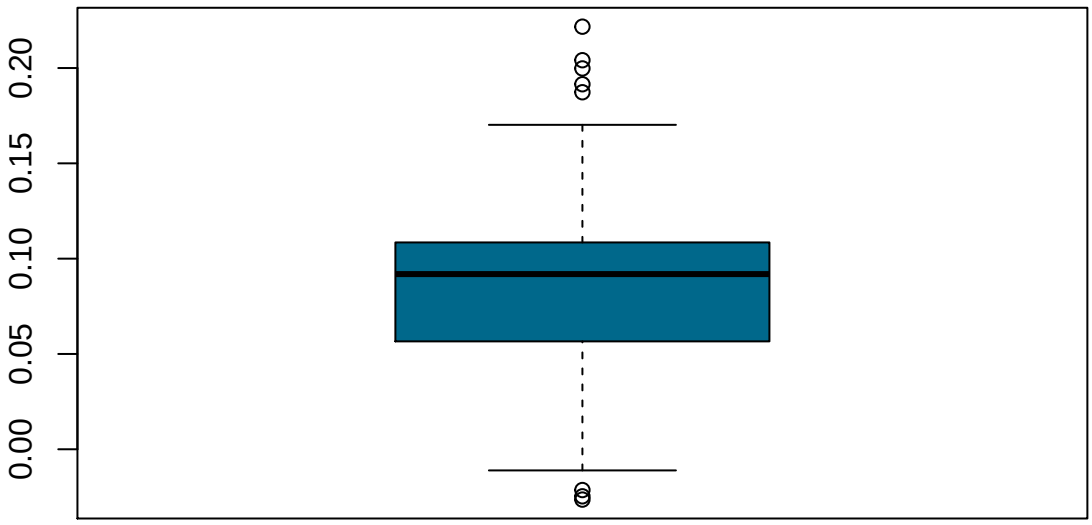
Median.Listing.Price.Yy change:

```
summary(listing100$Avg..Median.Listing.Price.Yy)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.02640  0.05665  0.09190  0.08586  0.10830  0.22170
```

Visual:

```
boxplot(listing100$Avg..Median.Listing.Price.Yy,col="deepskyblue4")
```



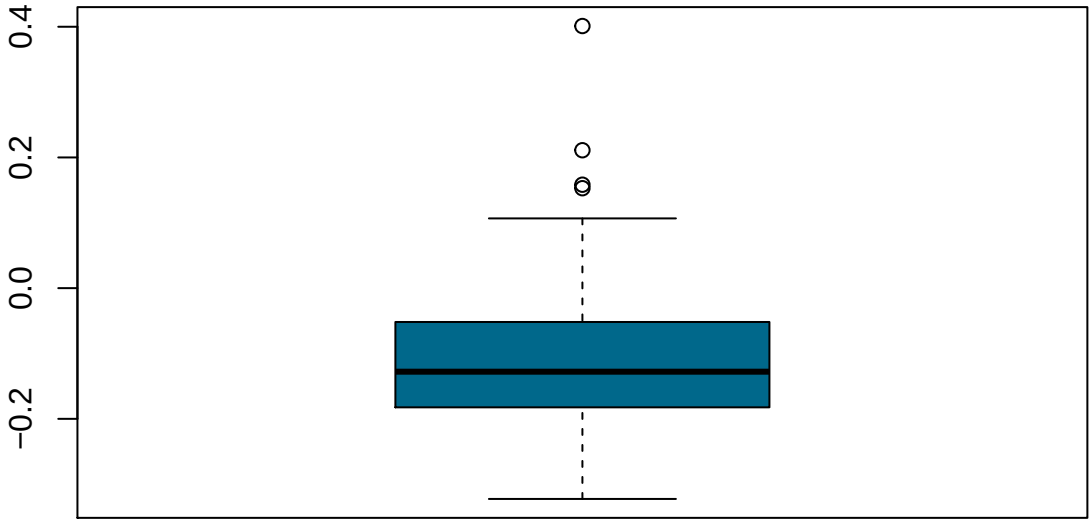
Active.Listing.Count.Yy change:

```
summary(listing100$Avg..Active.Listing.Count.Yy)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.32260 -0.18227 -0.12785 -0.10772 -0.05245  0.40110
```

Visual:

```
boxplot(listing100$Avg..Active.Listing.Count.Yy,col="deepskyblue4")
```



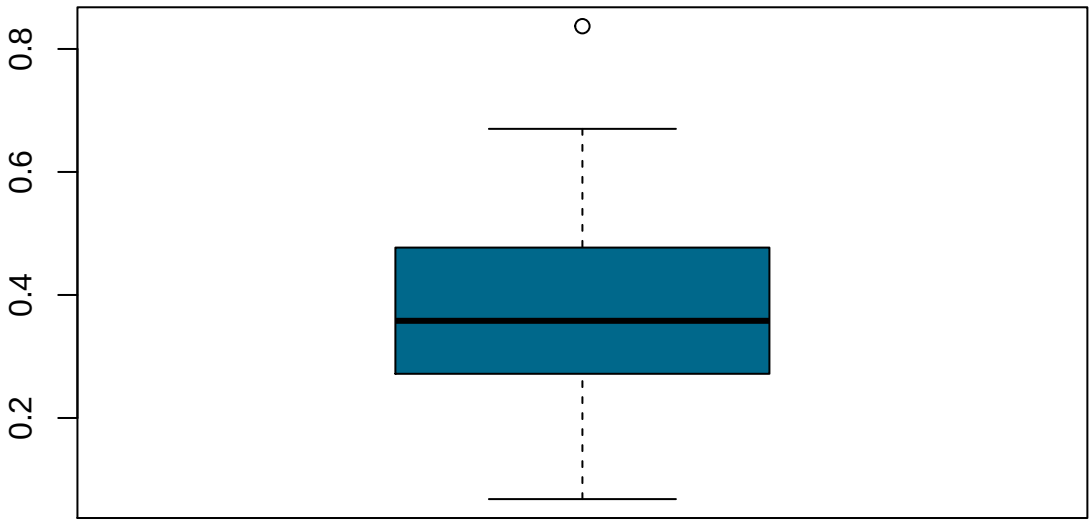
Ldpviews.Per.Property.Yy change:

```
summary(listing100$Avg..Ldpviews.Per.Property.Yy)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0681  0.2733  0.3580  0.3651  0.4749  0.8370
```

Visual:

```
boxplot(listing100$Avg..Ldpviews.Per.Property.Yy,col="deepskyblue4")
```



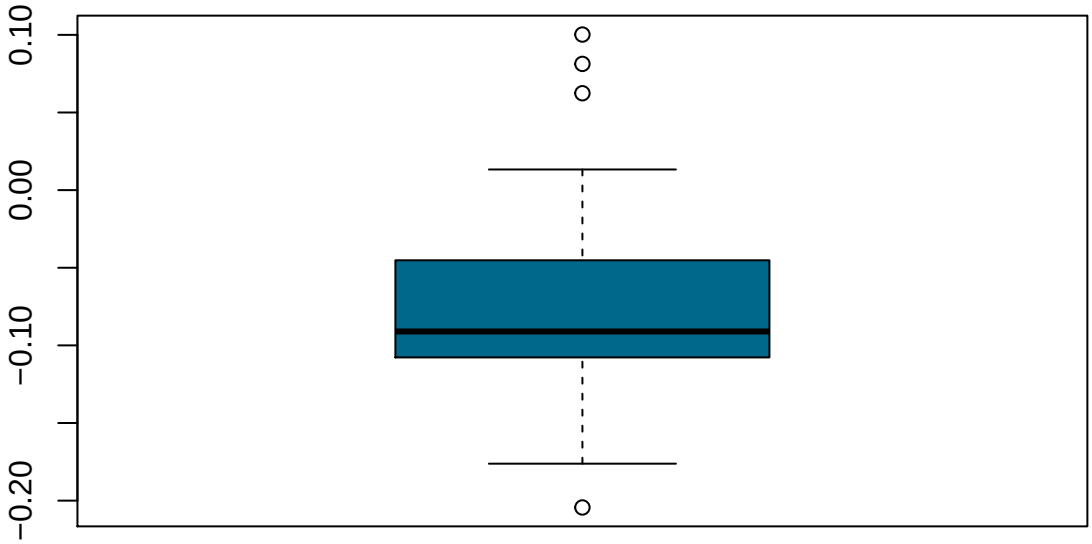
Median.Dom.Yy change:

```
summary(listing100$Avg..Median.Dom.Yy)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.20440 -0.10743 -0.09105 -0.07802 -0.04580  0.10020
```

Visual:

```
boxplot(listing100$Avg..Median.Dom.Yy,col="deepskyblue4")
```



Median.Listing.Price :

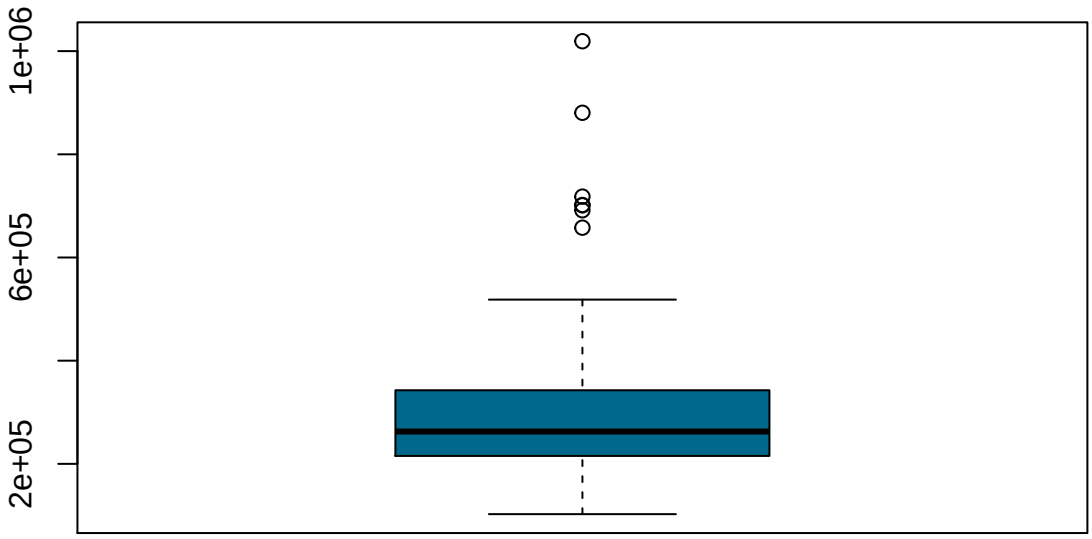
```
summary(listing100$Avg..Median.Listing.Price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```

```
##  102551  215647  262713  304097  342162 1019182
```

Visual:

```
boxplot(listing100$Avg..Median.Listing.Price,col="deepskyblue4")
```



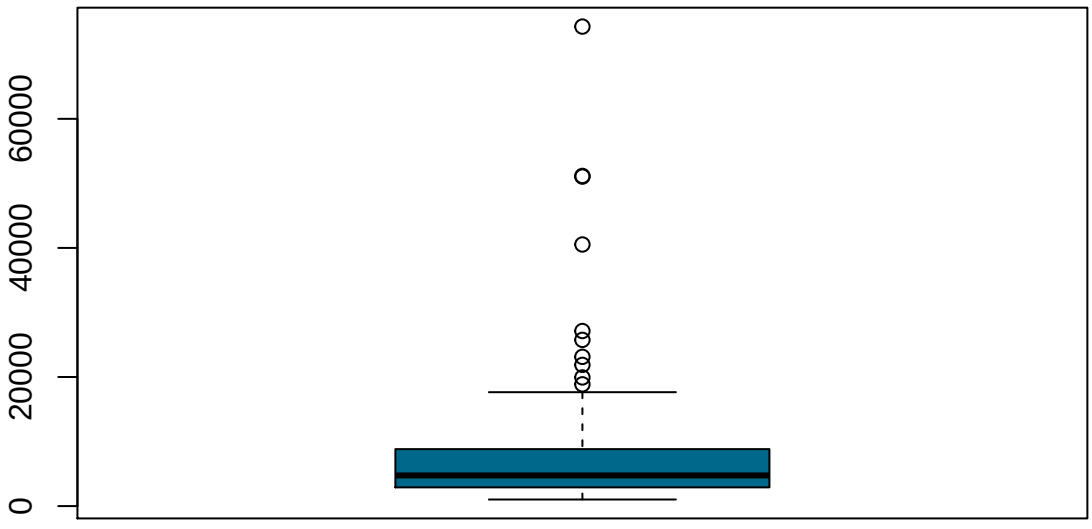
Active.Listing.Count :

```
summary(listing100$Avg..Active.Listing.Count)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1020    2901    4745    8403    8801   74292
```

Visual:

```
boxplot(listing100$Avg..Active.Listing.Count,col="deepskyblue4")
```



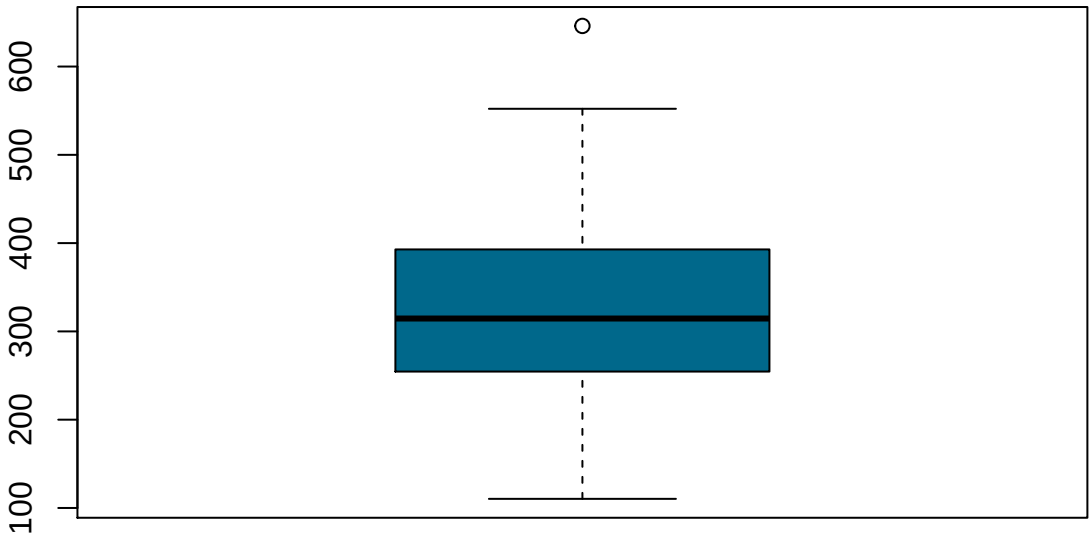
Ldpviews.Per.Property :

```
summary(listing100$Avg..Ldpviews.Per.Property)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   110.3   254.5   314.6   327.8   391.2   646.0
```

Visual:

```
boxplot(listing100$Avg..Ldpviews.Per.Property,col="deepskyblue4")
```



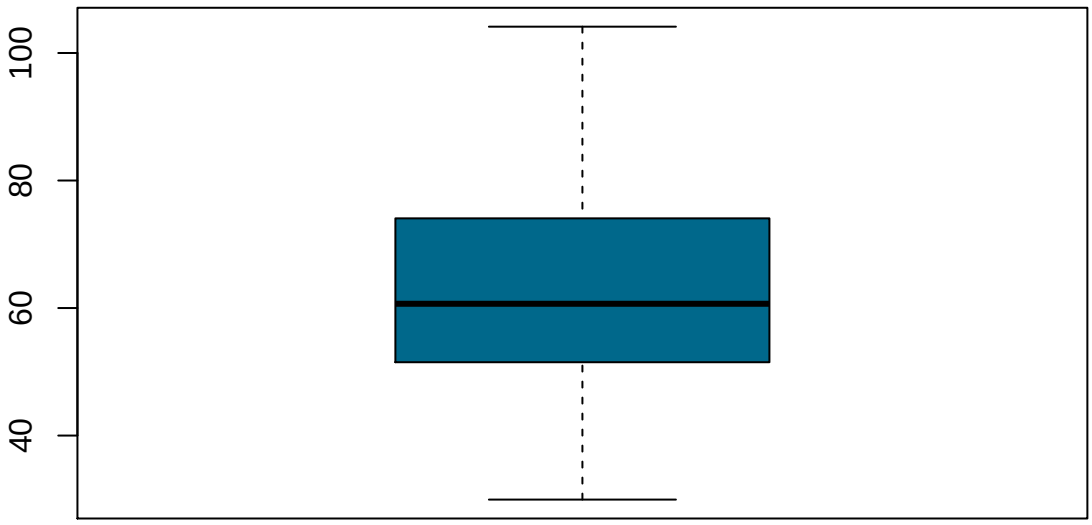
Median.Dom :

```
summary(listing100$Avg..Median.Dom)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   29.96   51.71   60.67   63.07   73.72  104.12
```

Visual:

```
boxplot(listing100$Avg..Median.Dom,col="deepskyblue4")
```



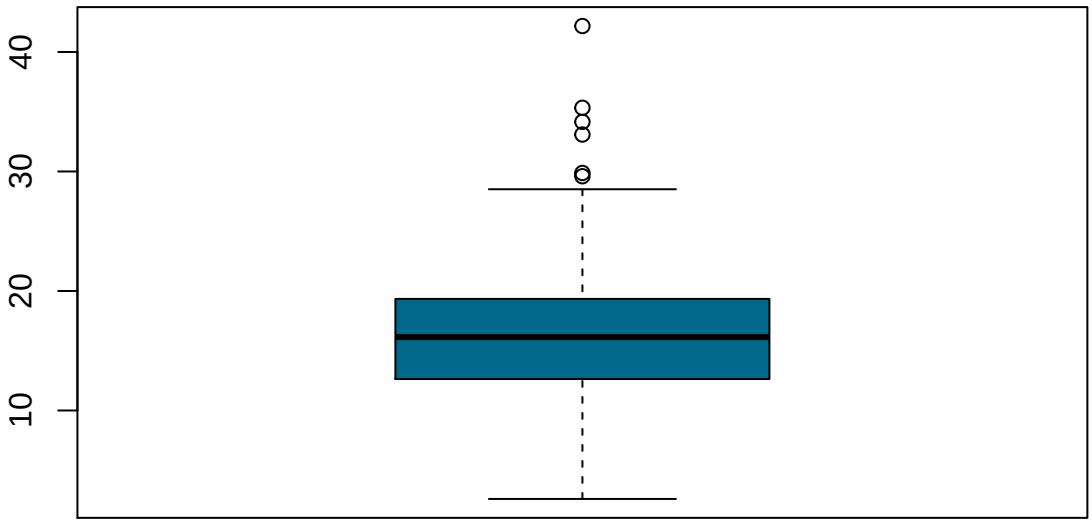
ActiveListingsper1000HH :

```
summary(listing100$Avg.ActiveListingsper1000HH)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.599  12.675  16.141  16.590  19.293  42.170
```

Visual:

```
boxplot(listing100$Avg.ActiveListingsper1000HH,col="deepskyblue4")
```



Now that we have decribed the varibles from the listing dataset, we can start performing various models on it.

Part 1 - Fitting models

STEP 1: LISTING DATA ANALYSIS

MODEL 1

We are going to analyze the listing data, followed by the economic data prior to running an analysis on both datasets combined. This will enable us to understand the nature of the variables in each dataset better prior to leading an overall analysis.

The first regression analysis we are going to perform includes all of the non-categorical listing variables listed above. The model is as follows:

```
fit1 <- lm(Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy + Avg..Ldpviews.Per.Property.Yy +

summary(fit1)
```

```
##
## Call:
## lm(formula = Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy +
##     Avg..Ldpviews.Per.Property.Yy + Avg..Median.Dom.Yy + Avg..Median.Listing.Price +
##     Avg..Active.Listing.Count + Avg..Ldpviews.Per.Property +
##     Avg..Median.Dom + Avg.ActiveListingsper1000HH, data = listing100)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.081178 -0.029250  0.000209  0.025380  0.095328
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   1.121e-01  4.133e-02   2.711 0.008017 **
## Avg..Active.Listing.Count.Yy  -1.992e-01  5.007e-02  -3.978 0.000139 ***
## Avg..Ldpviews.Per.Property.Yy  1.861e-02  3.946e-02   0.472 0.638334
## Avg..Median.Dom.Yy             1.115e-01  9.892e-02   1.127 0.262718
## Avg..Median.Listing.Price     -7.464e-08  3.047e-08  -2.450 0.016198 *
## Avg..Active.Listing.Count      5.672e-07  3.936e-07   1.441 0.152988
## Avg..Ldpviews.Per.Property     5.100e-05  4.843e-05   1.053 0.295009
## Avg..Median.Dom               -5.484e-04  3.875e-04  -1.415 0.160434
## Avg.ActiveListingsper1000HH   -5.995e-04  8.701e-04  -0.689 0.492610
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03879 on 91 degrees of freedom
## Multiple R-squared:  0.363,  Adjusted R-squared:  0.307
## F-statistic: 6.483 on 8 and 91 DF,  p-value: 1.172e-06
```

Conclusion about this regression: It seems that the initial model has a low adjusted R-squared and that very few variables are needed other than the intercept, Avg..Active.Listing.Count.Yy and Avg..Median.Listing.Price.

The overall F-test has a very small p-value which means that at least one variable is going to be needed.

Let's run a few diagnostics to ensure the model has appropriate data. The ncvTest will enable us to test for heteroskedasticity while the VIFs will enable us to test for multicollinearity.

```
install.packages("car")
```

```
## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/3.5'
## (as 'lib' is unspecified)
```

```
library(car)
```

```
## Loading required package: carData
```

```
ncvTest(fit1)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.1025332, Df = 1, p = 0.74881
```

The ncvTest has a very high p-value which indicates that there is no problem of heteroscedasticity, in other terms that the error terms have an equal variance.

```
vif(fit1)
```

```
##  Avg..Active.Listing.Count.Yy Avg..Ldpviews.Per.Property.Yy
##                      2.202165                      2.472944
##            Avg..Median.Dom.Yy     Avg..Median.Listing.Price
##                      1.743447                      1.490991
##     Avg..Active.Listing.Count    Avg..Ldpviews.Per.Property
##                      1.250565                      1.801861
##               Avg..Median.Dom   Avg.ActiveListingsper1000HH
##                      2.307929                      2.147613
```

The VIFs are all below a threshold of 10 which enables us to conclude that our data does not have any collinearity issues, which means relationships between the independent variables (the X's).

We can therefore more on to other analyses.

MODEL 2

We are going to run a stepwise regression to improve our prior model:

```
source("http://people.fas.harvard.edu/~mparzen/stat100/model_select.txt")
```

```
model.select(fit1, verbose = FALSE)
```

```
##
## Call:
## lm(formula = Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy +
##     Avg..Median.Listing.Price + Avg..Median.Dom, data = listing100)
##
## Coefficients:
##                   (Intercept)  Avg..Active.Listing.Count.Yy
##                     1.281e-01                     -2.072e-01
##    Avg..Median.Listing.Price               Avg..Median.Dom
##                    -5.840e-08                     -7.428e-04
```

The above model shows us that in the listing dataset Avg..Active.Listing.Count.Yy, Avg..Median.Listing.Price and Avg..Median.Dom are the variables the most needed in the model, other than categorical that we have not tested for yet.

MODEL 3

If we were to include the categorical variable, region, to the list of predictors, the new model would be as follows:

```
fit2 <- lm(Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy + Avg..Ldpviews.Per.Property.Yy +

summary(fit2, )
```

```
##
## Call:
## lm(formula = Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy +
##     Avg..Ldpviews.Per.Property.Yy + Avg..Median.Dom.Yy + Avg..Median.Listing.Price +
##     Avg..Active.Listing.Count + Avg..Ldpviews.Per.Property +
##     Avg..Median.Dom + Avg.ActiveListingsper1000HH + factor(Region),
##     data = listing100)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.080671 -0.025676 -0.005296  0.023279  0.078227
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   9.131e-02  4.405e-02   2.073 0.041123 *
## Avg..Active.Listing.Count.Yy -1.713e-01  4.943e-02  -3.466 0.000818 ***
## Avg..Ldpviews.Per.Property.Yy  5.067e-02  4.023e-02   1.259 0.211244
## Avg..Median.Dom.Yy             6.922e-02  9.909e-02   0.699 0.486639
## Avg..Median.Listing.Price     -1.852e-08  3.462e-08  -0.535 0.594103
## Avg..Active.Listing.Count      5.141e-07  3.930e-07   1.308 0.194291
## Avg..Ldpviews.Per.Property     5.914e-05  4.812e-05   1.229 0.222326
## Avg..Median.Dom               -3.709e-04  4.338e-04  -0.855 0.394879
## Avg.ActiveListingsper1000HH   -6.691e-04  8.497e-04  -0.787 0.433127
## factor(Region)Northeast       -3.732e-02  1.350e-02  -2.763 0.006964 **
## factor(Region)South           -1.377e-02  1.207e-02  -1.141 0.256966
## factor(Region)West            -3.801e-02  1.491e-02  -2.549 0.012531 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03722 on 88 degrees of freedom
## Multiple R-squared:  0.4327, Adjusted R-squared:  0.3618
## F-statistic: 6.103 on 11 and 88 DF,  p-value: 2.413e-07
```

The model 3 shows that by adding the categorical variables region, the variables needed change and we now need the intercept, Avg..Active.Listing.Count.Yy, factor(Region)Northeast and factor(Region)West . The adjusted R-squared is higher than model 1 and the SSE is lower, which means that our overall prediction is better.

MODEL 3

We are now going to run stepwise regression again, including the categorical variables:

```
model.select(fit2, verbose = FALSE)
```

```
##
## Call:
```

```
## lm(formula = Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy +
##     Avg..Median.Dom + factor(Region), data = listing100)
##
## Coefficients:
##                   (Intercept)  Avg..Active.Listing.Count.Yy
##                      0.127781                     -0.211819
##               Avg..Median.Dom       factor(Region)Northeast
##                     -0.000657                     -0.036186
##          factor(Region)South            factor(Region)West
##                     -0.018675                     -0.042346
```

Now, let us look at the suggested above predictors individually:

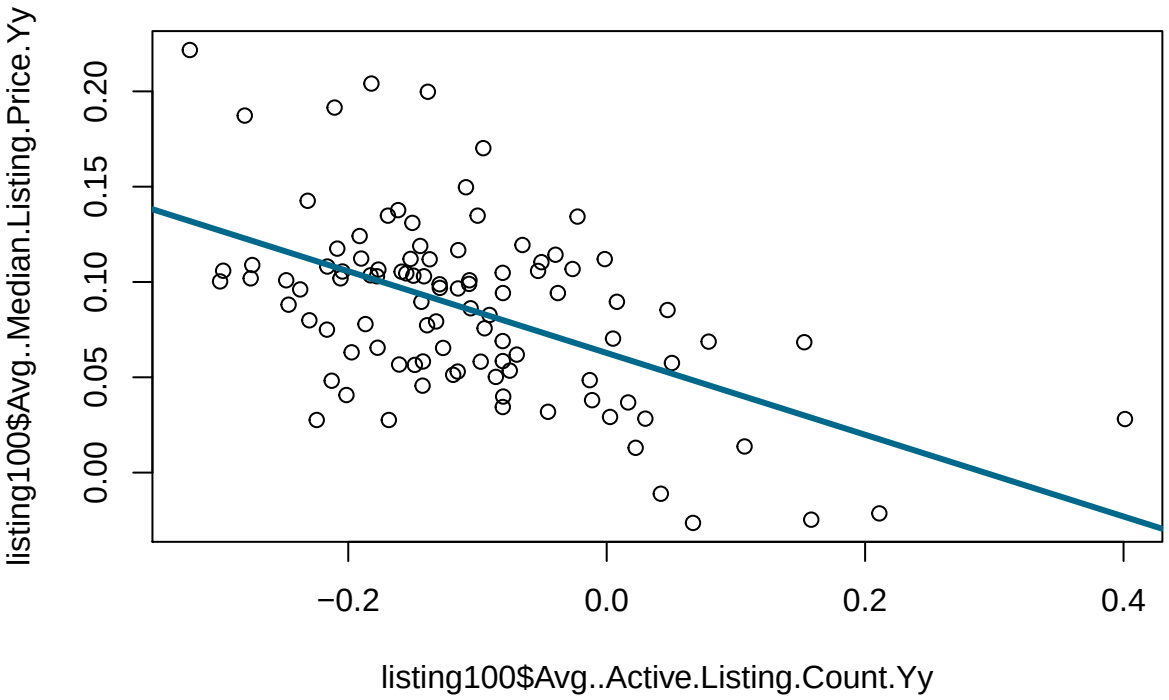
```
fit3 <- lm(Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy, data=listing100)

summary(fit3)
```

```
##
## Call:
## lm(formula = Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy,
##     data = listing100)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.083288 -0.027609 -0.000271  0.024880  0.107357
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   0.062759   0.005435  11.546  < 2e-16 ***
## Avg..Active.Listing.Count.Yy -0.214478   0.034503  -6.216 1.25e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03966 on 98 degrees of freedom
## Multiple R-squared:  0.2828, Adjusted R-squared:  0.2755
## F-statistic: 38.64 on 1 and 98 DF,  p-value: 1.246e-08
```

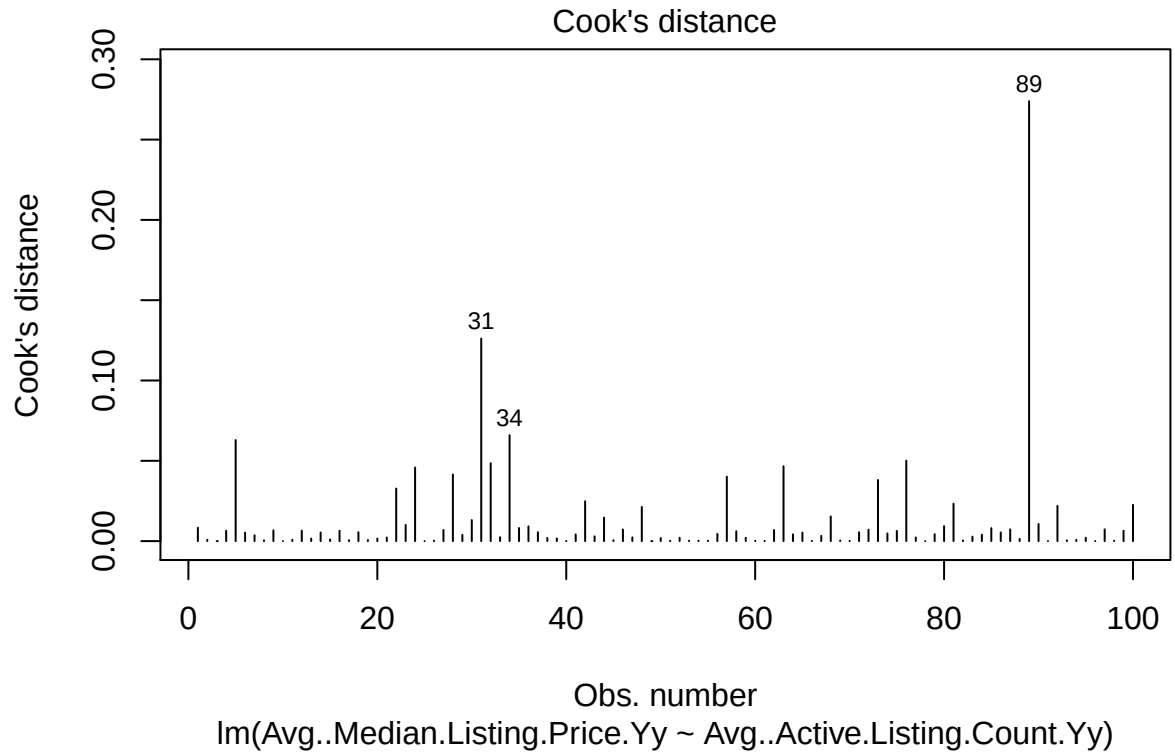
Scatterplot and regression line

```
plot(listing100$Avg..Active.Listing.Count.Yy, listing100$Avg..Median.Listing.Price.Yy)
abline(fit3, col="deepskyblue4", lwd=3)
```



We are going to look at Cook's distance to search for influential points that would bias our models:

```
plot(fit3,4)
```



Let us list the two longest on Cook's distance:

```
listing100[c(31,89),1:4]
```

```
##    RankHH Cbsacode                          Cbsatitle  Region
```

```
## 31     31    18140                       Columbus, OH Midwest
## 89     89    37340 Palm Bay-Melbourne-Titusville, FL   South
```

Let us remove those two points.

```
listing98<-listing100[!(listing100$RankHH == 31 | listing100$RankHH == 89),]
dim(listing98)
```

```
## [1] 98 13
```

MODEL 4

With a cleaner set of data, we can now run our 4th model:

```
fit4 <- lm(Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy, data=listing98)

summary(fit4)
```

```
##
## Call:
## lm(formula = Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy,
##     data = listing98)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.082662 -0.026881 -0.000348  0.025570  0.108599
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   0.060526   0.005765  10.499  < 2e-16 ***
## Avg..Active.Listing.Count.Yy -0.221642   0.038328  -5.783 9.15e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03862 on 96 degrees of freedom
## Multiple R-squared:  0.2583, Adjusted R-squared:  0.2506
## F-statistic: 33.44 on 1 and 96 DF,  p-value: 9.15e-08
```

This 4th model has a slightly improved adjusted R-squared and a slightly better SSE, making it a better model.

Partial conclusion: Because we think that not adding the categorical variable into our initial models would be a mistake, we can conclude with confidence that our best model for the listing dataset is model #4.

The next step in our process is to add the economic variables onto the prior data and create a more refined model with all the data that we have. After doing this, we are going to make initial conclusions on our findings in order to answer our theoretical question.

STEP 2: ALL DATA ANALYSIS

We first need to combine both datasets in order to run further analysis:

```
listing_eco_98 <- merge(listing98, economy, by="RankHH")
dim(listing_eco_98)
```

```
## [1] 98 43
```

We now have a long list of columns. The list is as follows:

```
names(listing_eco_98)
```

```
##  [1] "RankHH"                          "Cbsacode"
##  [3] "Cbsatitle"                       "Region.x"
##  [5] "Avg..Median.Listing.Price.Yy"    "Avg..Active.Listing.Count.Yy"
##  [7] "Avg..Ldpviews.Per.Property.Yy"   "Avg..Median.Dom.Yy"
##  [9] "Avg..Median.Listing.Price"       "Avg..Active.Listing.Count"
## [11] "Avg..Ldpviews.Per.Property"      "Avg..Median.Dom"
## [13] "Avg.ActiveListingsper1000HH"     "Cbsa.Code"
## [15] "Cbsa.Title"                      "Region.y"
## [17] "End.Of.2016.Household"           "End.Of.2017.Household"
## [19] "End.Of.2016.Job"                 "End.Of.2017.Job"
## [21] "Income.2017"                     "Income.2016"
## [23] "Unemployment.Rate.2017"          "Unemployment.Rate.2016"
## [25] "Buy.Pct.2017"                    "Buy.Pct.2016"
## [27] "HH_yoy"                          "JOB_yoy"
## [29] "INC_yoy"                         "UNEMP_yoy"
## [31] "BUYPCT_yoy"                      "Home.Ownership.2017"
## [33] "OwnOccHH2017"                    "Sale.Px.Recovery"
## [35] "Sale.Price.Yoy"                  "Sale.Px.Recovery..Msa1."
## [37] "Sale.Price.Yoy..Msa1."           "Total.Starts"
## [39] "Total.Starts.Recovery"           "Total.Starts.Yoy"
## [41] "Total.Starts..Msa1."             "Total.Starts.Recovery..Msa1."
## [43] "Total.Starts.Yoy..Msa1."
```

MODEL 5

We are going to run regression with all our data.

```
fit5=lm(Avg..Median.Listing.Price.Yy ~Avg..Active.Listing.Count.Yy + Avg..Ldpviews.Per.Property.Yy + Avg
 + Total.Starts..Msa1. + Total.Starts.Recovery..Msa1. + Total.Starts.Yoy..Msa1., data = listing_eco_98)
summary(fit5)
```

```
##
## Call:
## lm(formula = Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy +
##     Avg..Ldpviews.Per.Property.Yy + Avg..Median.Dom.Yy + Avg..Median.Listing.Price +
##     Avg..Active.Listing.Count + Avg..Ldpviews.Per.Property +
##     Avg..Median.Dom + Region.x + Avg.ActiveListingsper1000HH +
##     End.Of.2016.Household + End.Of.2017.Household + End.Of.2016.Job +
##     End.Of.2017.Job + Income.2017 + Income.2016 + Unemployment.Rate.2017 +
##     Unemployment.Rate.2016 + Buy.Pct.2017 + Buy.Pct.2016 + HH_yoy +
##     JOB_yoy + INC_yoy + UNEMP_yoy + BUYPCT_yoy + Home.Ownership.2017 +
##     OwnOccHH2017 + Sale.Px.Recovery + Sale.Price.Yoy + Sale.Px.Recovery..Msa1. +
##     Sale.Price.Yoy..Msa1. + Total.Starts + Total.Starts.Recovery +
##     Total.Starts.Yoy + Total.Starts..Msa1. + Total.Starts.Recovery..Msa1. +
##     Total.Starts.Yoy..Msa1., data = listing_eco_98)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.046923 -0.012973 -0.000697  0.010279  0.046754
```

```
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -2.961e+00  9.096e-01  -3.255  0.00195 **
## Avg..Active.Listing.Count.Yy  -1.507e-01  5.038e-02  -2.991  0.00416 **
## Avg..Ldpviews.Per.Property.Yy -2.677e-02  3.506e-02  -0.763  0.44844
## Avg..Median.Dom.Yy            -4.739e-02  9.398e-02  -0.504  0.61610
## Avg..Median.Listing.Price      2.723e-07  1.426e-07   1.909  0.06152 .
## Avg..Active.Listing.Count      9.748e-07  1.093e-06   0.892  0.37652
## Avg..Ldpviews.Per.Property     1.930e-06  3.979e-05   0.049  0.96149
## Avg..Median.Dom               -1.850e-04  4.270e-04  -0.433  0.66659
## Region.xNortheast              1.462e-03  1.419e-02   0.103  0.91830
## Region.xSouth                 -1.067e-02  1.549e-02  -0.689  0.49390
## Region.xWest                  -1.933e-02  1.827e-02  -1.058  0.29471
## Avg.ActiveListingsper1000HH   -1.159e-03  9.065e-04  -1.279  0.20641
## End.Of.2016.Household         -4.125e-07  7.952e-07  -0.519  0.60604
## End.Of.2017.Household          4.069e-07  7.723e-07   0.527  0.60044
## End.Of.2016.Job               -1.835e-07  2.095e-07  -0.876  0.38491
## End.Of.2017.Job                1.918e-07  2.077e-07   0.924  0.35970
## Income.2017                   -1.991e-05  1.088e-05  -1.830  0.07263 .
## Income.2016                    1.892e-05  1.172e-05   1.615  0.11213
## Unemployment.Rate.2017        -1.797e-02  2.529e-02  -0.710  0.48043
## Unemployment.Rate.2016         1.172e-02  2.328e-02   0.504  0.61654
## Buy.Pct.2017                  -1.135e+00  5.663e-01  -2.005  0.04993 *
## Buy.Pct.2016                   8.774e-01  5.000e-01   1.755  0.08487 .
## HH_yoy                         8.169e-01  6.704e-01   1.218  0.22827
## JOB_yoy                        2.107e-02  2.972e-01   0.071  0.94375
## INC_yoy                        1.378e+00  6.586e-01   2.092  0.04109 *
## UNEMP_yoy                      2.693e-02  1.114e-01   0.242  0.80984
## BUYPCT_yoy                     8.011e-01  1.649e-01   4.859 1.02e-05 ***
## Home.Ownership.2017            6.824e-02  1.040e-01   0.656  0.51450
## OwnOccHH2017                  -1.100e-08  5.372e-08  -0.205  0.83851
## Sale.Px.Recovery               2.135e-01  1.713e-01   1.246  0.21796
## Sale.Price.Yoy                -2.140e-01  1.726e-01  -1.240  0.22027
## Sale.Px.Recovery..Msa1.       -1.971e-01  1.753e-01  -1.124  0.26579
## Sale.Price.Yoy..Msa1.          2.768e-01  1.548e-01   1.789  0.07920 .
## Total.Starts                   1.306e-06  1.771e-06   0.738  0.46376
## Total.Starts.Recovery         -6.638e-02  4.675e-02  -1.420  0.16124
## Total.Starts.Yoy               1.710e-02  2.237e-02   0.764  0.44792
## Total.Starts..Msa1.           -1.836e-06  2.309e-06  -0.795  0.42977
## Total.Starts.Recovery..Msa1.   5.792e-02  4.665e-02   1.242  0.21965
## Total.Starts.Yoy..Msa1.       -7.372e-03  1.586e-02  -0.465  0.64389
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02361 on 55 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.8311, Adjusted R-squared:  0.7144
## F-statistic: 7.123 on 38 and 55 DF,  p-value: 5.178e-11
```

We are going to run some diagnostics, particularly as it relates to heteroskedasticity and multicollinearity:

```
ncvTest(fit5)
```

```
## Non-constant Variance Score Test
```

```
## Variance formula: ~ fitted.values
## Chisquare = 2.62936, Df = 1, p = 0.1049
```

No heteroskedasticity in this 5th model either.

```
vif(fit5)
```

```
##                                      GVIF Df GVIF^(1/(2*Df))
## Avg..Active.Listing.Count.Yy  4.300110e+00  1        2.073671
## Avg..Ldpviews.Per.Property.Yy 4.927225e+00  1        2.219735
## Avg..Median.Dom.Yy            4.015640e+00  1        2.003906
## Avg..Median.Listing.Price     8.568957e+01  1        9.256866
## Avg..Active.Listing.Count     2.577369e+01  1        5.076780
## Avg..Ldpviews.Per.Property    2.735730e+00  1        1.654004
## Avg..Median.Dom               6.861996e+00  1        2.619541
## Region.x                      1.445305e+02  3        2.290832
## Avg.ActiveListingsper1000HH   6.076315e+00  1        2.465018
## End.Of.2016.Household         1.128288e+05  1      335.899989
## End.Of.2017.Household         1.064770e+05  1      326.308167
## End.Of.2016.Job               1.352281e+04  1      116.287611
## End.Of.2017.Job               1.356072e+04  1      116.450484
## Income.2017                   2.861675e+03  1       53.494624
## Income.2016                   2.756343e+03  1       52.500884
## Unemployment.Rate.2017        1.228641e+02  1       11.084406
## Unemployment.Rate.2016        1.238882e+02  1       11.130507
## Buy.Pct.2017                  5.101570e+02  1       22.586655
## Buy.Pct.2016                  4.204852e+02  1       20.505735
## HH_yoy                        7.073373e+00  1        2.659581
## JOB_yoy                       5.120130e+00  1        2.262770
## INC_yoy                       5.792830e+01  1        7.611064
## UNEMP_yoy                     3.396311e+01  1        5.827788
## BUYPCT_yoy                    1.658351e+01  1        4.072285
## Home.Ownership.2017           3.989669e+00  1        1.997415
## OwnOccHH2017                  1.580867e+02  1       12.573253
## Sale.Px.Recovery              1.743917e+02  1       13.205745
## Sale.Price.Yoy                1.684019e+01  1        4.103680
## Sale.Px.Recovery..Msa1.       1.626496e+02  1       12.753416
## Sale.Price.Yoy..Msa1.         9.687793e+00  1        3.112522
## Total.Starts                  6.135480e+01  1        7.832931
## Total.Starts.Recovery         1.680822e+01  1        4.099783
## Total.Starts.Yoy              7.144729e+00  1        2.672963
## Total.Starts..Msa1.           8.781783e+01  1        9.371117
## Total.Starts.Recovery..Msa1.  1.953294e+01  1        4.419609
## Total.Starts.Yoy..Msa1.       5.734050e+00  1        2.394588
```

Some of the VIFs are definitely above 100, which indicates collinearity in our dataset.

MODEL 6

We are going to run a stepwise regression on the prior model to indicate which variables are needed in the model. We can then compare the SSE and adjusted R-squared:

```
model.select(fit5, verbose = FALSE)
```

```
##
## Call:
## lm(formula = Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy +
##     End.Of.2016.Household + End.Of.2017.Household + Income.2017 +
##     Unemployment.Rate.2017 + INC_yoy + BUYPCT_yoy + Sale.Price.Yoy..Msa1.,
##     data = listing_eco_98)
##
## Coefficients:
##                   (Intercept)  Avg..Active.Listing.Count.Yy
##                    -7.393e-01                     -1.431e-01
##         End.Of.2016.Household         End.Of.2017.Household
##                    -1.046e-06                      1.053e-06
##                  Income.2017        Unemployment.Rate.2017
##                    -8.863e-07                     -1.206e-02
##                       INC_yoy                     BUYPCT_yoy
##                     3.326e-01                      5.385e-01
##         Sale.Price.Yoy..Msa1.
##                     1.163e-01
```

In this stepwise refression, model 6 indicates that only 8 variables are needed, and are as follows: Avg..Active.Listing.Count.Yy, End.Of.2016.Household, End.Of.2017.Household, Income.2017, Unemployment.Rate.2017, INC_yoy, BUYPCT_yoy and Sale.Price.Yoy..Msa1

We are going to run a final model with only those variables and compare it to model 5.

```
fit6=lm(Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy+Income.2017+End.Of.2016.Household+E
summary(fit6)
```

```
##
## Call:
## lm(formula = Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy +
##     Income.2017 + End.Of.2016.Household + End.Of.2017.Household +
##     Unemployment.Rate.2017 + INC_yoy + BUYPCT_yoy + Sale.Price.Yoy..Msa1.,
##     data = listing_eco_98)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.065484 -0.015624 -0.001434  0.015933  0.050899
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -7.393e-01  1.423e-01  -5.195 1.38e-06 ***
## Avg..Active.Listing.Count.Yy -1.431e-01  2.724e-02  -5.255 1.08e-06 ***
## Income.2017                   -8.863e-07  2.286e-07  -3.877 0.000207 ***
## End.Of.2016.Household         -1.046e-06  2.847e-07  -3.674 0.000417 ***
## End.Of.2017.Household          1.053e-06  2.847e-07   3.697 0.000386 ***
## Unemployment.Rate.2017        -1.206e-02  2.465e-03  -4.892 4.67e-06 ***
## INC_yoy                        3.326e-01  1.043e-01   3.188 0.002005 **
## BUYPCT_yoy                     5.385e-01  5.143e-02  10.470  < 2e-16 ***
## Sale.Price.Yoy..Msa1.          1.163e-01  5.102e-02   2.280 0.025090 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02335 on 85 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.7446, Adjusted R-squared:  0.7206
```

```
## F-statistic: 30.98 on 8 and 85 DF,  p-value: < 2.2e-16
```

In comparison to all our prior models, model 6 is by far the model that has the highest adjusted R-squared and the lowest SSE, making it our most reliable model. We are going to run more diagnostics on the model 6 to ensure the data we are model respects the four assumptions of linear regression.

```
ncvTest(fit6)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.02115518, Df = 1, p = 0.88436
```

Model 6 does not present any problem of heteroscedasticity

```
vif(fit6)
```

```
## Avg..Active.Listing.Count.Yy                  Income.2017
##                     1.284570                    1.291537
##        End.Of.2016.Household        End.Of.2017.Household
##                  14775.873121                14788.271561
##       Unemployment.Rate.2017                     INC_yoy
##                     1.192591                    1.484936
##                   BUYPCT_yoy        Sale.Price.Yoy..Msa1.
##                     1.649256                    1.075973
```

The VIFs analysis highlights that End.Of.2016.Household and End.Of.2017.Household have multicollinearity issues. Therefore, we are going to drop end of 2017 households and use the 2016 data for the purpose of modeling.

MODEL 7

```
fit7=lm(Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy+Income.2017+End.Of.2016.Household+U
summary(fit7)
```

```
##
## Call:
## lm(formula = Avg..Median.Listing.Price.Yy ~ Avg..Active.Listing.Count.Yy +
##     Income.2017 + End.Of.2016.Household + Unemployment.Rate.2017 +
##     INC_yoy + BUYPCT_yoy + Sale.Price.Yoy..Msa1., data = listing_eco_98)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.069413 -0.014695 -0.002218  0.014610  0.070964
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -6.996e-01  1.520e-01  -4.603 1.43e-05 ***
## Avg..Active.Listing.Count.Yy -1.260e-01  2.875e-02  -4.383 3.29e-05 ***
## Income.2017                   -8.224e-07  2.441e-07  -3.369  0.00113 **
## End.Of.2016.Household          6.350e-09  2.657e-09   2.390  0.01906 *
## Unemployment.Rate.2017       -1.314e-02  2.621e-03  -5.014 2.83e-06 ***
## INC_yoy                        3.022e-01  1.114e-01   2.713  0.00805 **
## BUYPCT_yoy                     5.389e-01  5.508e-02   9.783 1.25e-15 ***
## Sale.Price.Yoy..Msa1.          1.109e-01  5.462e-02   2.030  0.04543 *
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02502 on 86 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.7035, Adjusted R-squared:  0.6794
## F-statistic: 29.16 on 7 and 86 DF,  p-value: < 2.2e-16
```

```
vif(fit7)
```

```
## Avg..Active.Listing.Count.Yy                  Income.2017
##                     1.247451                    1.284154
##        End.Of.2016.Household       Unemployment.Rate.2017
##                     1.122256                    1.175726
##                      INC_yoy                   BUYPCT_yoy
##                     1.475729                    1.649248
##        Sale.Price.Yoy..Msa1.
##                     1.075076
```

This model does not have any issues of collinearity anymore. We are going to run the ncvTest one last time to ensure that this model does not have any heteroskedasticity issue either:

```
ncvTest(fit7)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.5032978, Df = 1, p = 0.47805
```

We are here confirming that model 7 also does not present any heteroskedasticity issue.

CONCLUSION

Our statistical analysis has revealed the 7 variables that explain the average median listing price year over year. The average active listing count year over year is a first variable. It indicates the amount of listings available as a snapshot at a moment in time of the supply of properties for sale. The income for the year is an obvious variable that defines the purchasing power of a household to buy a house. Unemployment rate, a very talked-about metrics in the US at the moment, defines the share of the population without an employment. It goes without saying that the higher the employment rate the greater the share of the population that has access to financial viability to purchase a property as well as getting financing.

Our analysis highlighted that, unlike what was thought initially, region (our only categorical variable in the set) does not affect the average median listing price year over year, nor does the number of online views onto Realtor.com website. Our initial assumption was that most online views, the higher the average median listing price year over year. In fact, we proved that a fair amount of viewership is simply related to looking at real estate in a passive manner as opposed to looking at real estate solely prior to a purchase.