Housing EDA

David Gerard

2019-02-11

Researchers were interested in predicting residential home sales prices in a Midwestern city as a function of various characteristics of the home and surrounding property. Data on 522 transactions were obtained for home sales during the year 2002. The 13 variables are

- `Price`: Sales price of residence (in dollars)
- `Area`: Finished area of residence (in square feet)
- `Bed`: Total number of bedrooms in residence
- `Bath`: Total number of bathrooms in residence
- `AC`: 1 = presence of air conditioning, 0 = absence of air conditioning
- `Garage`: Number of cars that a garage will hold
- `Pool`: 1 = presence of a pool, 0 = absence of a pool
- `Year`: Year property was originally constructed
- `Quality`: Index for quality of construction. `High`, `Medium`, or `Low`.
- `Style`: Categorical variable indicating architectural style
- `Lot`: Lot size (in square feet)
- `Highway`: 1 = highway adjacent, 0 = highway not adjacent.

The data are available in “estate.csv” at https://dcgerard.github.io/stat_412_612/data/estate.csv.

Perform an exploratory data analysis to come up with some hypotheses. Some suggested ways to focus your research:

- Which variables are categorical? Which are quantitative?
- Change the values of the categorical variables to something more informative.
- What variables are marginally associated with price? Use plots and summary statistics.
- What variables are marginally associated with each other? Use plots and summary statistics.
- If a variable is marginally associated with price, are there some other variables that could explain that association? Use plots and summary statistics.
- Does there appear to be any discrete groupings of houses?
- Are there any unusual observations?
- What transformations should you perform to make associations more linear?
- Try making new variables based on existing variables.
- What variables should be discretized (or have values aggregated) because there are too few values and/or the association seems discrete?
- If you know linear regression, try out the `lm()` and `step()` functions to choose a tentative model.

Load required packages

```r
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------------------------------------
```

```
## v ggplot2 3.1.0     v purrr   0.3.0
## v tibble  2.0.1     v dplyr   0.7.8
## v tidyr   0.8.2     v stringr 1.3.1
## v readr   1.3.1     v forcats 0.3.0
## -- Conflicts ------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(GGally) ## for pairs plot
```

```
##
## Attaching package: 'GGally'

## The following object is masked from 'package:dplyr':
##
##     nasa
```

```
theme_set(theme_bw())
```

Load Data into R

```
estate <- read_csv("../../data/estate.csv")
```

```
## Parsed with column specification:
## cols(
##   Price = col_double(),
##   Area = col_double(),
##   Bed = col_double(),
##   Bath = col_double(),
##   AC = col_double(),
##   Garage = col_double(),
##   Pool = col_double(),
##   Year = col_double(),
##   Quality = col_character(),
##   Style = col_double(),
##   Lot = col_double(),
##   Highway = col_double()
## )
```

```
head(estate)
```

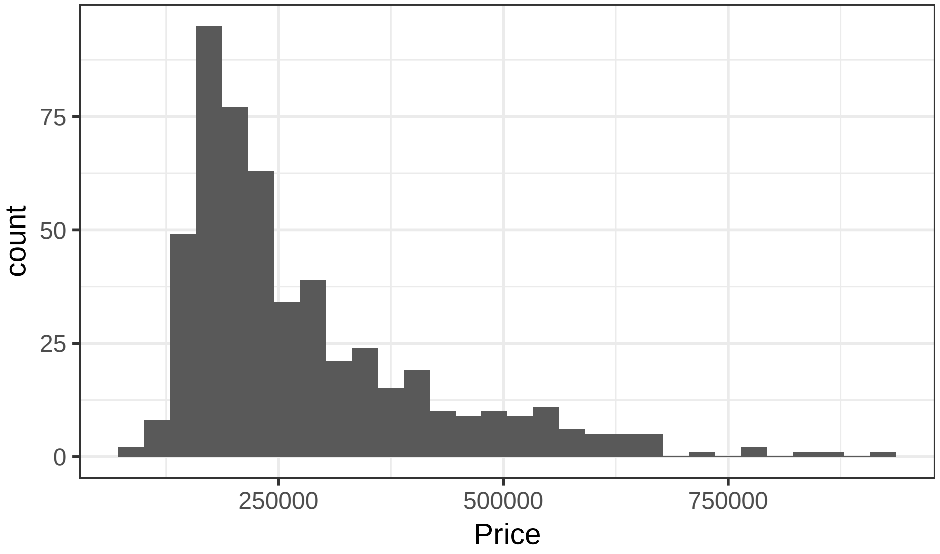
```
## # A tibble: 6 x 12
##    Price  Area   Bed  Bath    AC Garage  Pool  Year Quality Style   Lot
##    <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <chr>   <dbl> <dbl>
## 1 360000  3032     4     4     1      2     0  1972 Medium      1 22221
## 2 340000  2058     4     2     1      2     0  1976 Medium      1 22912
## 3 250000  1780     4     3     1      2     0  1980 Medium      1 21345
## 4 205500  1638     4     2     1      2     0  1963 Medium      1 17342
## 5 275500  2196     4     3     1      2     0  1968 Medium      7 21786
## 6 248000  1966     4     3     1      5     1  1972 Medium      1 18902
## # ... with 1 more variable: Highway <dbl>
```

I'm going to change the obvious variables that should be factors to factors.

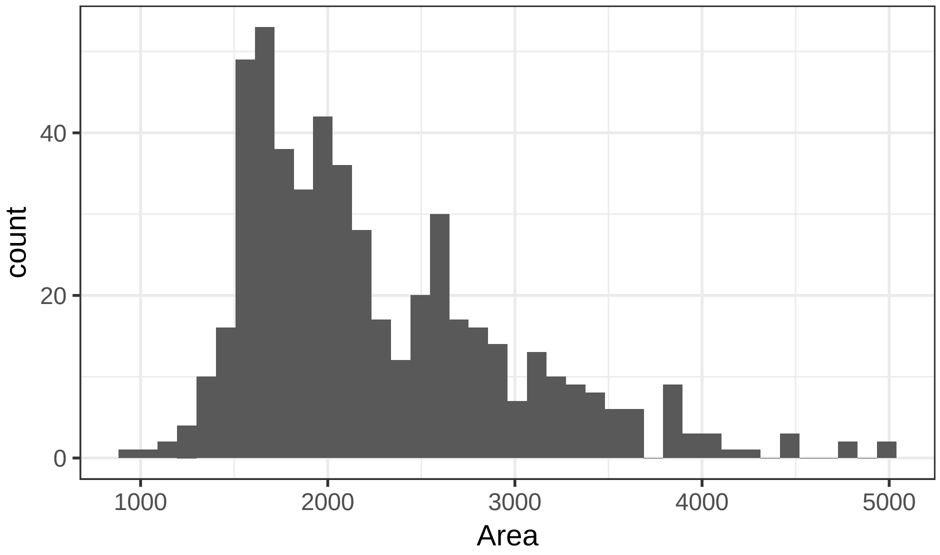
```
estate %>%
  mutate(AC = recode(AC, "1" = "AC", "0" = "noAC"),
         Pool = recode(Pool, "1" = "Pool", "0" = "noPool"),
         Style = as.factor(Style),
         Highway = recode(Highway, "1" = "Highway", "0" = "noHighway")) ->
  estate
```

Univariate Distributions

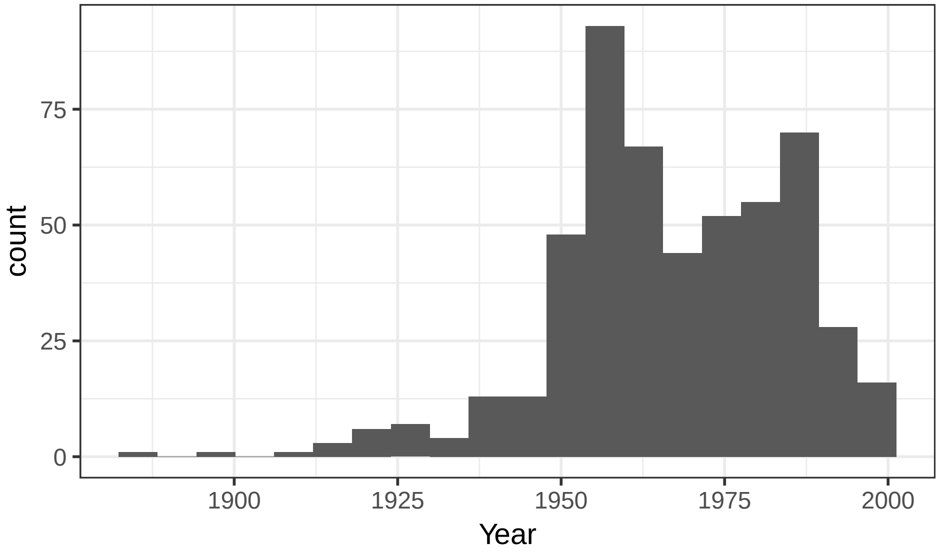
```
ggplot(estate, aes(x = Price)) +
  geom_histogram(bins = 30)
```



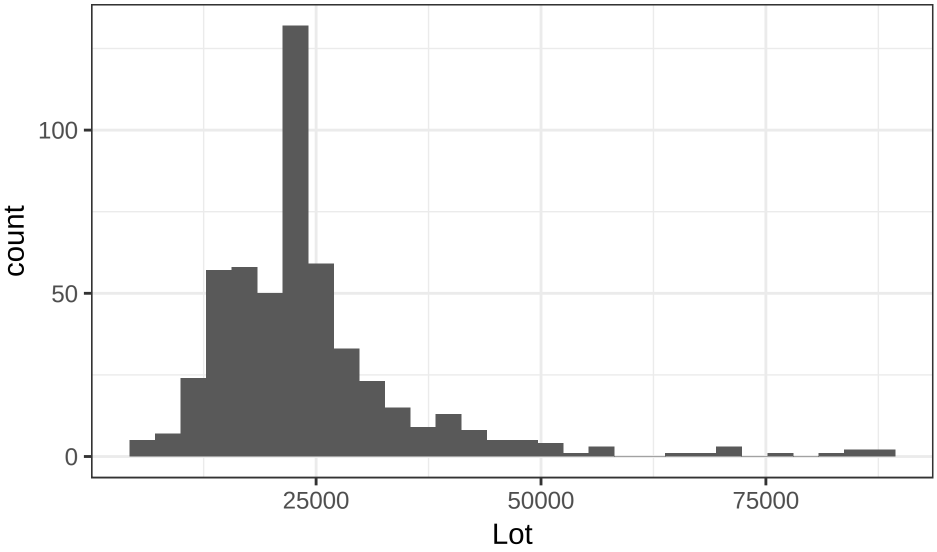
```
## Two bumps in Area followed by a long tail
ggplot(estate, aes(x = Area)) +
  geom_histogram(bins = 40)
```



```r
ggplot(estate, aes(x = Year)) +
  geom_histogram(bins = 20)
```



```r
ggplot(estate, aes(x = Lot)) +
  geom_histogram(bins = 30)
```



```
## mostly 3 and 4 bedroom houses, but there is a 0 bedroom house
## Is that a studio?
table(estate$Bed)
```

```
##
##   0   1   2   3   4   5   6   7
##   1   9  64 202 179  52  12   3
```

```
## A 0 bathroom house??? Is that the same house?
table(estate$Bath)
```

```
##
##   0   1   2   3   4   5   6   7
##   1  71 171 175  84  17   1   2
```

```
## Let's look at that unit
estate %>%
  filter(Bath == 0)
```

```
## # A tibble: 1 x 12
##    Price  Area   Bed  Bath AC    Garage Pool   Year Quality Style   Lot
##    <dbl> <dbl> <dbl> <dbl> <chr>  <dbl> <chr> <dbl> <chr>   <fct> <dbl>
## 1 528750  2129     0     0 AC         3 noPo~  1992 High    1     37414
## # ... with 1 more variable: Highway <chr>
```

```
## It's price is on the high end for having no bathroom!
## (92nd percentile)
estate %>%
  filter(Bath == 0) %>%
  select(Price) %>%
  c() ->
  weird_house_price
mean(estate$Price < weird_house_price)
```

```
## [1] 0.9215
```

```
## I woul dkeep in mind removing that house if I was to go on and do a
## Linear regression
```

```
table(estate$AC)
```

```
##
##    AC noAC
##   434   88
```

```
## One garage holds 7 cars?
table(estate$Garage)
```

```
##
##    0   1   2   3   4   5   7
##    7  52 353 106   2   1   1
```

```
table(estate$Pool)
```

```
##
## noPool   Pool
##    486     36
```

```
table(estate$Quality)
```

```
##
##   High    Low Medium
##     68    164    290
```

```
table(estate$Style)
```

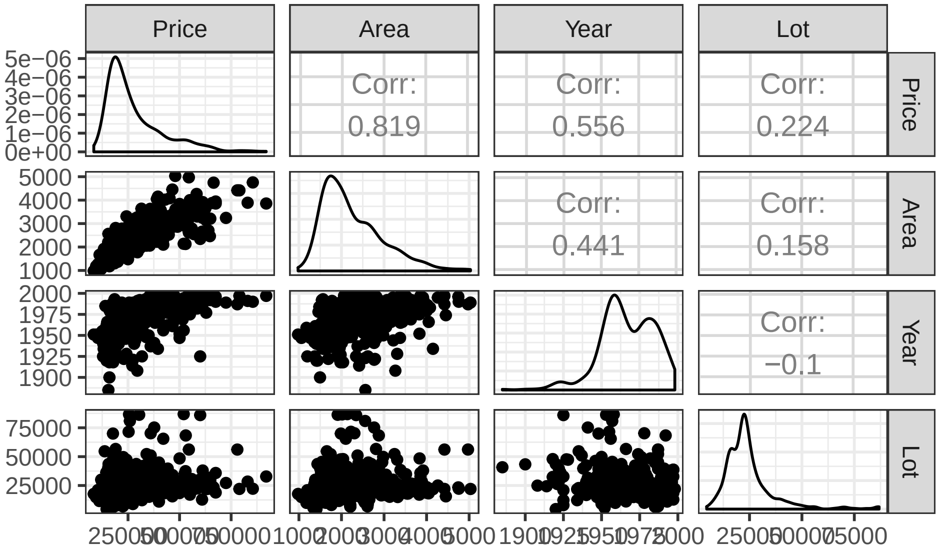
```
##
##   1   2   3   4   5   6   7   9  10  11
## 214  58  64  11  18  18 136   1   1   1
```

```
## Very few houses on a highway. I would be
## Careful about inferences there
table(estate$Highway)
```

```
##
##   Highway noHighway
##        11       511
```

Look at bivariate associations

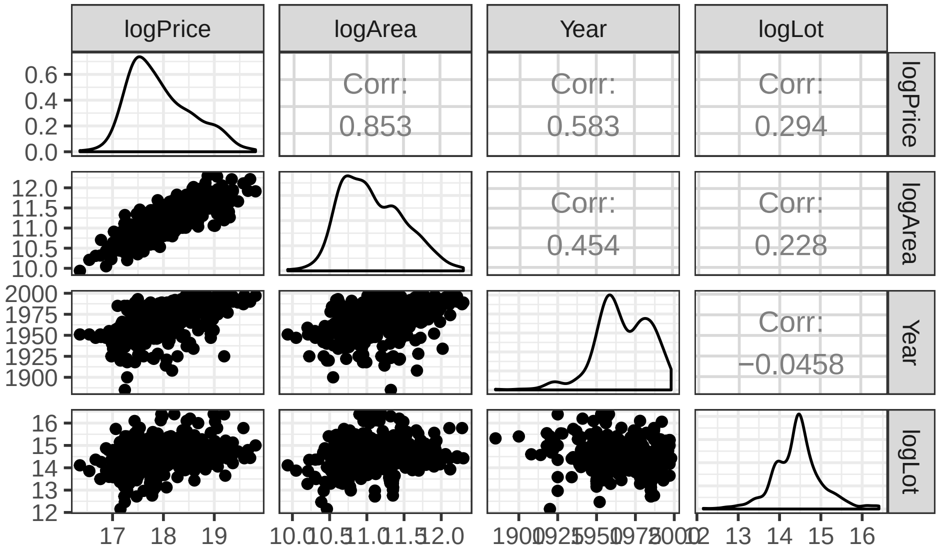
```
estate %>%
  select(Price, Area, Year, Lot) %>%
  ggpairs()
```



It seems that we can log price and area and lot pretty safely.

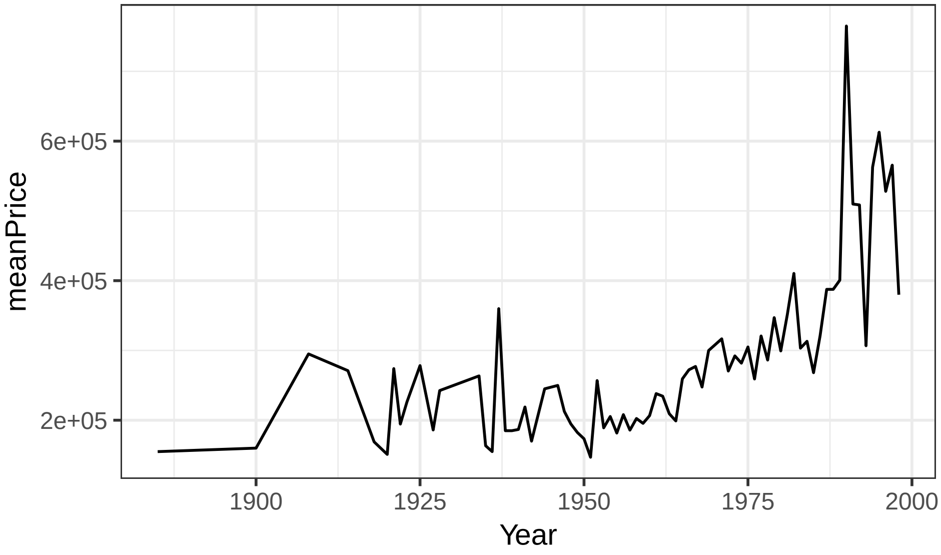
```
estate %>%
  mutate(logPrice = log2(Price), logArea = log2(Area), logLot = log2(Lot)) ->
  estate

estate %>%
  select(logPrice, logArea, Year, logLot) %>%
  ggpairs()
```



It seems that area has the strongest relationship to price

```
estate %>%
  group_by(Year) %>%
  summarize(meanPrice = mean(Price)) %>%
  ggplot(aes(x = Year, y = meanPrice)) +
  geom_line()
```



```
summary(estate$Year)
```

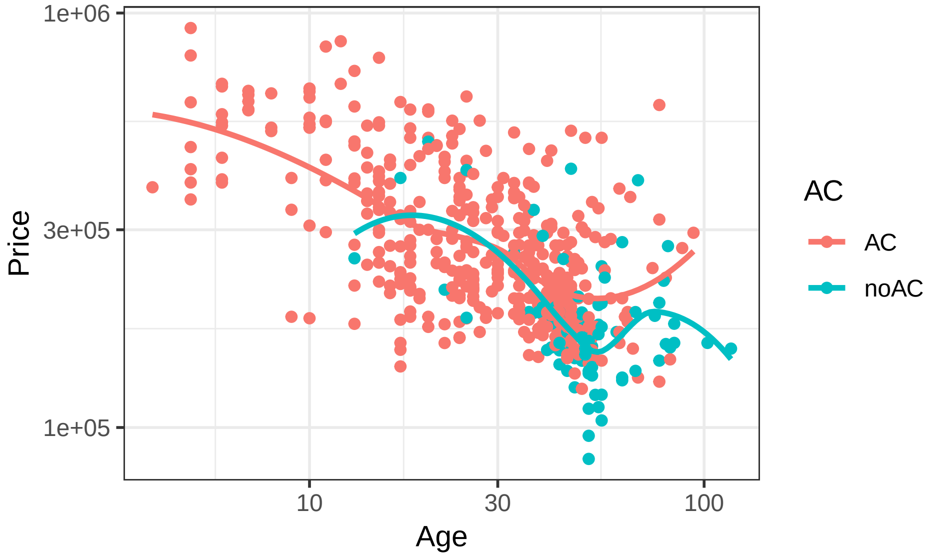
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1885    1956    1966    1967    1981    1998
```

```
## Define Age of hours at sale
estate %>%
  mutate(Age = 2002 - Year) ->
  estate
```

Q: Does anything variable get more important with the age of the house?

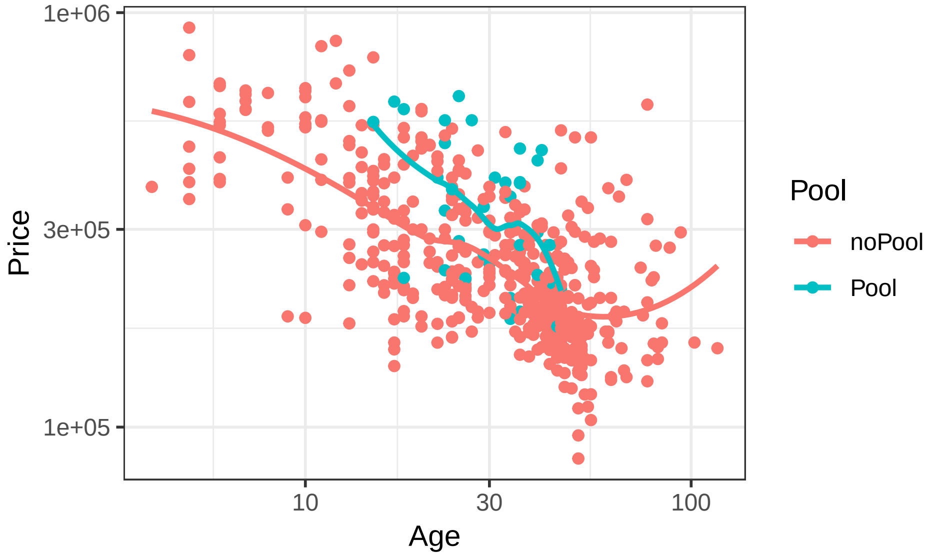
```
## Most no-ac houses are older. And once you adjust for age.
## But it still seems that there is an AC effect, which is particularly
## strong for older houses.
ggplot(estate, aes(x = Age, y = Price, color = AC)) +
  geom_point() +
  geom_smooth(se = FALSE) +
  scale_x_log10() +
  scale_y_log10()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

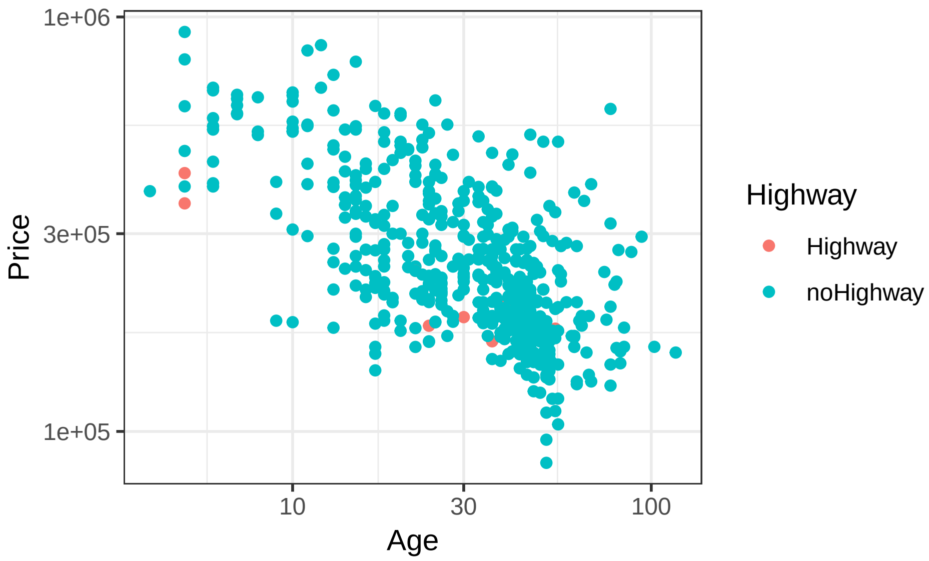


```
## There seems to be an additive pool effect
ggplot(estate, aes(x = Age, y = Price, color = Pool)) +
  geom_point() +
  geom_smooth(se = FALSE) +
  scale_x_log10() +
  scale_y_log10()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



```
## The houses near the highway almost always have a lower price
ggplot(estate, aes(x = Age, y = Price, color = Highway)) +
  geom_point() +
  scale_x_log10() +
  scale_y_log10()
```

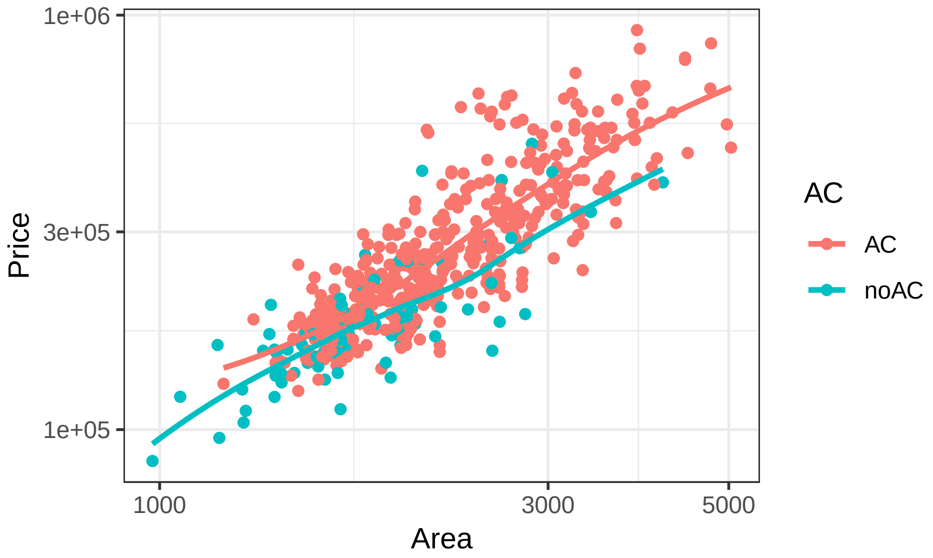


```
estate %>%
  mutate(logAge = log2(Age)) ->
  estate
```

Redo coloring with Area

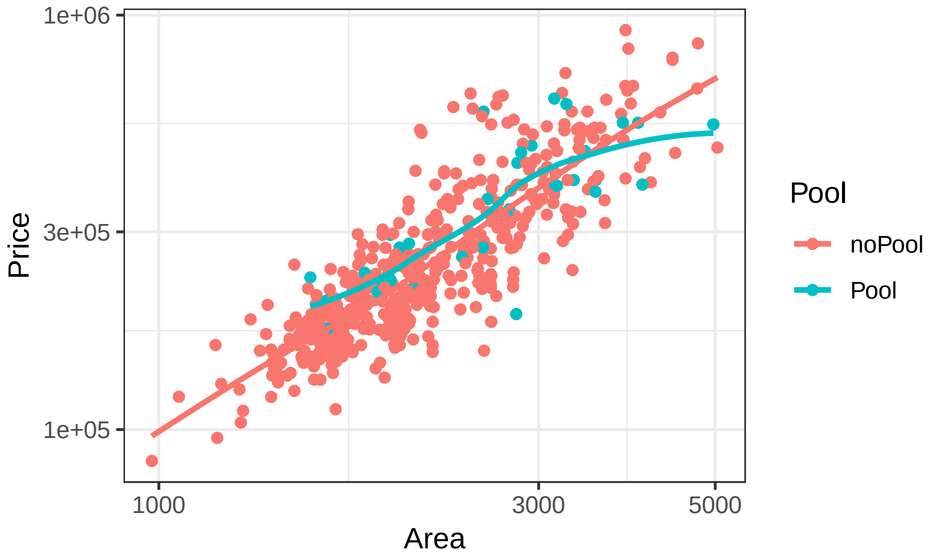
```
## Most no-ac houses are older. And once you adjust for age.
## But it still seems that there is an AC effect, which is particularly
## strong for older houses.
ggplot(estate, aes(x = Area, y = Price, color = AC)) +
  geom_point() +
  geom_smooth(se = FALSE) +
  scale_x_log10() +
  scale_y_log10()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

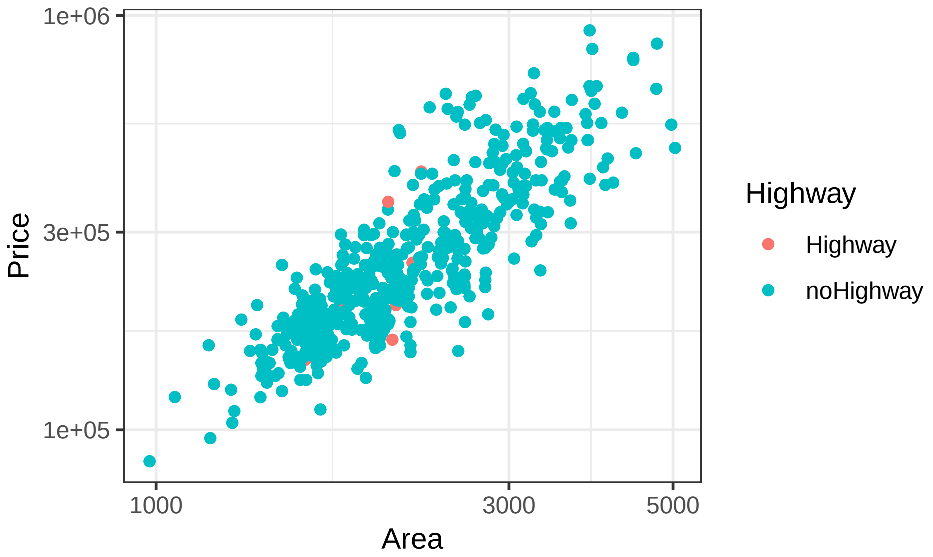


```
## There seems to be an additive pool effect
ggplot(estate, aes(x = Area, y = Price, color = Pool)) +
  geom_point() +
  geom_smooth(se = FALSE) +
  scale_x_log10() +
  scale_y_log10()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

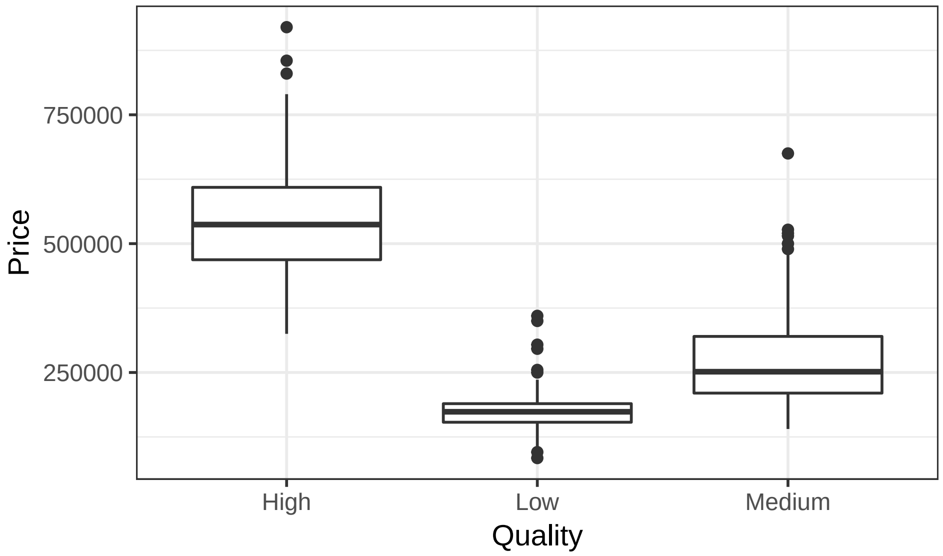


```
## The houses near the highway almost always have a lower price
ggplot(estate, aes(x = Area, y = Price, color = Highway)) +
  geom_point() +
  scale_x_log10() +
  scale_y_log10()
```

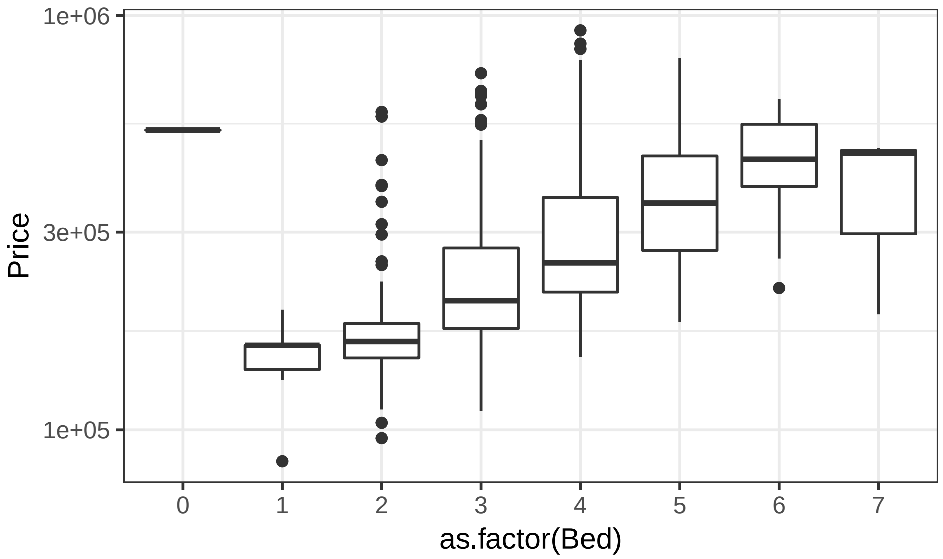


Look at the price by those other categories

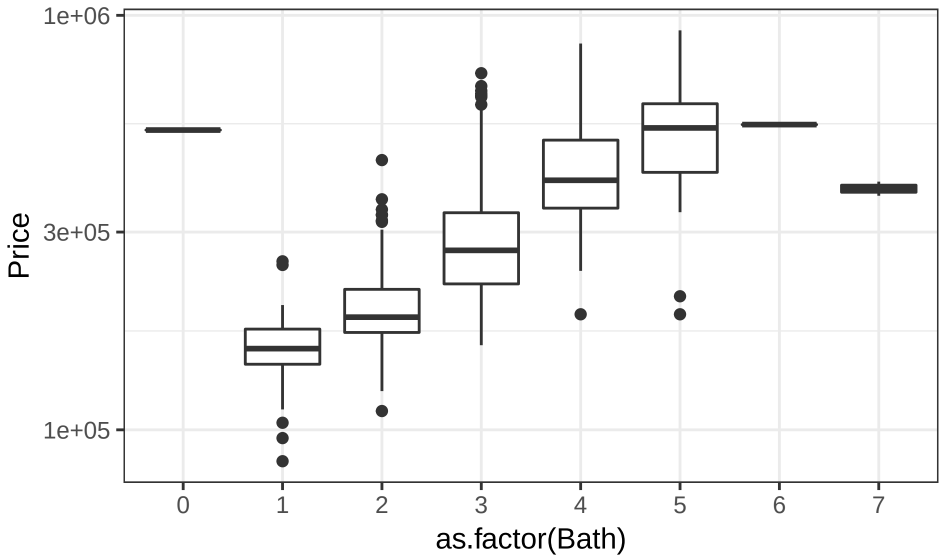
```
estate %>%
  ggplot(aes(x = Quality, y = Price)) +
  geom_boxplot()
```



```
## Seems that we could treat bed as a
## Quantitative variable if we got
## rid of that 0 house
estate %>%
  ggplot(aes(x = as.factor(Bed), y = Price)) +
  geom_boxplot() +
  scale_y_log10()
```



```
## I'm going to marge 5 bath into 5 and above,
## But still treat it as a quantitative variable
estate %>%
  ggplot(aes(x = as.factor(Bath), y = Price)) +
  geom_boxplot() +
  scale_y_log10()
```



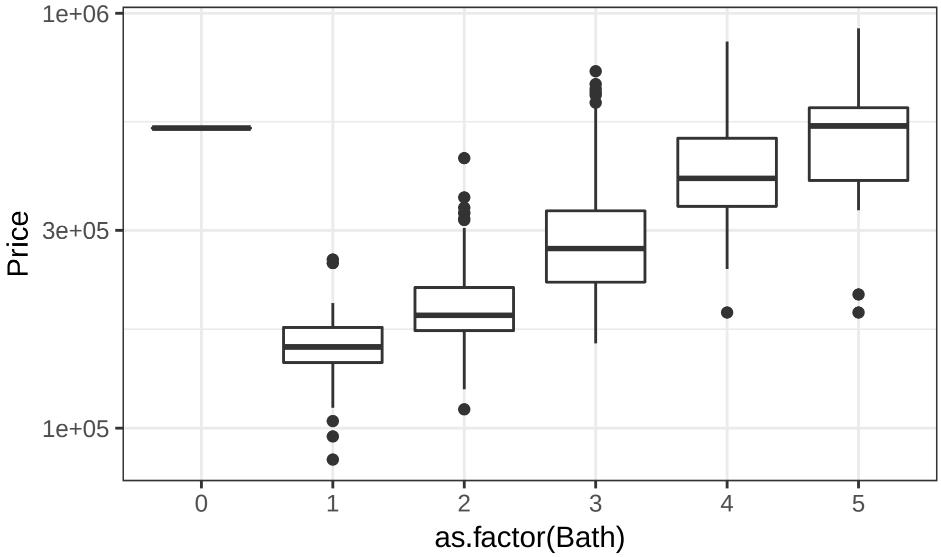
```
estate %>%
  mutate(Bath = recode(Bath, `6` = 5, `7` = 5)) ->
  estate

table(estate$Bath)
```

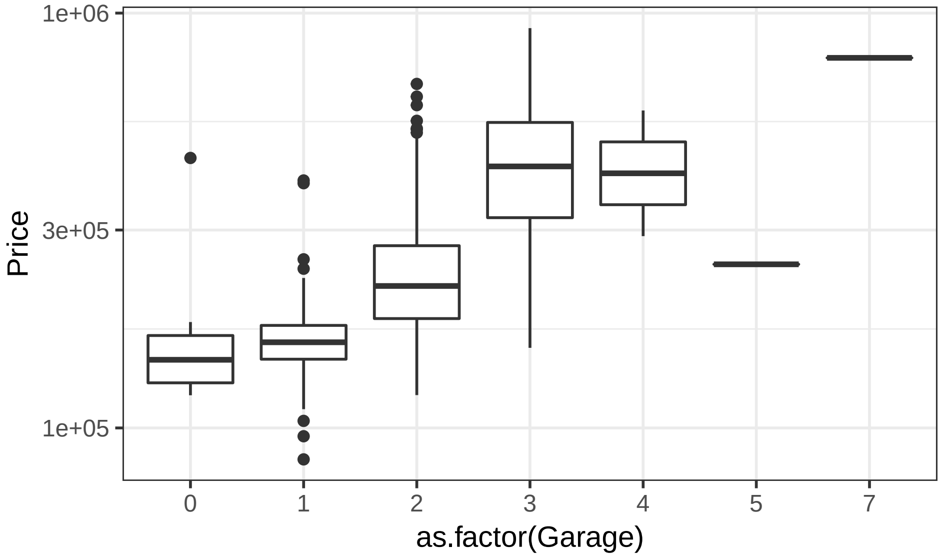
```
##
```

```
##   0   1   2   3   4   5
##   1  71 171 175  84  20
```

```
estate %>%
  ggplot(aes(x = as.factor(Bath), y = Price)) +
  geom_boxplot() +
  scale_y_log10()
```



```
## Garage also looks pretty linear
estate %>%
  ggplot(aes(x = as.factor(Garage), y = Price)) +
  geom_boxplot() +
  scale_y_log10()
```



Summary of interesting Observations

- There seems to be a bathroom saturation effect. Having more than 5 doesn't help you that much.
- Most quantitative relationships with price are exponential in nature. In other words, a multiplicative difference in area corresponds to a multiplicative difference in price. A multiplicative difference in age corresponds to a multiplicative difference in price.
- A lot of the discrete quantities (number of garages, number of baths, number of beds) can be treated as a linear relationship with log Price. So if you add one more bedroom, it results in some multiplicative change in price.
- There is one observation with no bed, no bath, but is super pricey. My guess is that this is an empty lot on some prime real estate. I would exclude this from any model and state clearly that our analysis is for homes with at least one bedroom.

Some exploratory forward/backward linear model stuff

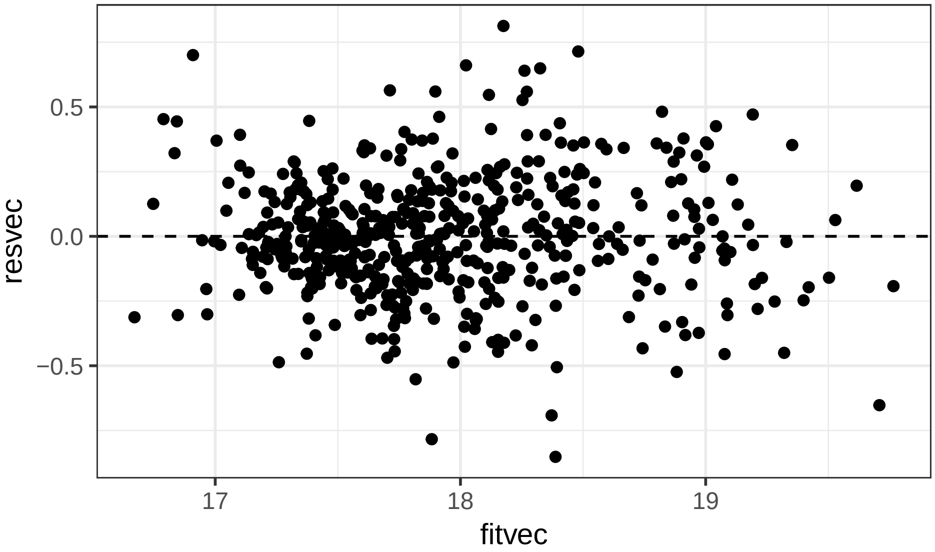
```
estate %>%
  filter(Bath != 0) ->
  estate_sub
lmfull <- lm(logPrice ~ Bed + Bath + AC + Garage + Pool +
               Quality + Style + Highway + logArea + logLot + logAge,
             data = estate_sub)

sout <- step(lmfull)
```

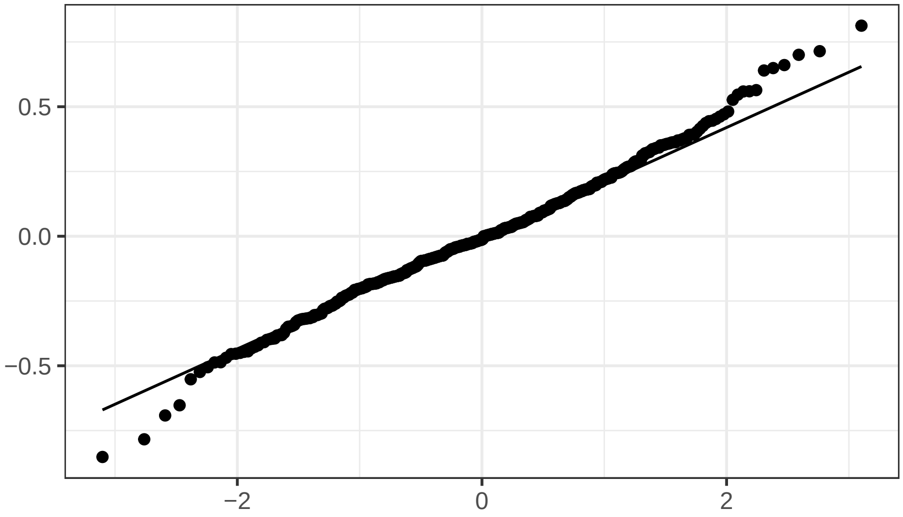
```
## Start:  AIC=-1488
## logPrice ~ Bed + Bath + AC + Garage + Pool + Quality + Style +
##     Highway + logArea + logLot + logAge
##
##           Df Sum of Sq  RSS   AIC
## - Bed      1      0.07 27.7 -1488
## <none>                 27.7 -1488
## - AC       1      0.12 27.8 -1487
## - Garage   1      0.15 27.8 -1487
## - Highway  1      0.23 27.9 -1485
## - Pool     1      0.42 28.1 -1482
## - Bath     1      0.80 28.5 -1475
## - Style    9      2.00 29.7 -1469
## - logLot   1      2.13 29.8 -1451
## - Quality  2      3.74 31.4 -1426
## - logAge   1      4.06 31.7 -1418
## - logArea  1     10.42 38.1 -1323
##
## Step:  AIC=-1488
## logPrice ~ Bath + AC + Garage + Pool + Quality + Style + Highway +
##     logArea + logLot + logAge
##
##           Df Sum of Sq  RSS   AIC
## <none>                 27.7 -1488
## - AC       1      0.14 27.9 -1488
## - Garage   1      0.15 27.9 -1488
## - Highway  1      0.22 27.9 -1486
## - Pool     1      0.42 28.1 -1483
## - Bath     1      1.00 28.7 -1472
## - Style    9      1.98 29.7 -1470
## - logLot   1      2.16 29.9 -1451
## - Quality  2      3.68 31.4 -1428
## - logAge   1      4.00 31.7 -1420
## - logArea  1     11.43 39.1 -1311
```

```
resvec <- resid(sout)
fitvec <- fitted(sout)

## Residuals look pretty awesome
qplot(fitvec, resvec) +
  geom_hline(yintercept = 0, linetype = "dashed")
```



```
qplot(sample = resvec, geom = "qq") +
  geom_qq_line()
```



```
coefvec <- coef(sout)
confintmat <- confint(sout)
sumlm <- summary(sout)
sumlm
```

```
##
## Call:
## lm(formula = logPrice ~ Bath + AC + Garage + Pool + Quality +
##     Style + Highway + logArea + logLot + logAge, data = estate_sub)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8520 -0.1517 -0.0128  0.1365  0.8128
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        8.4897     0.6033   14.07  < 2e-16
## Bath               0.0765     0.0180    4.25  2.6e-05
## ACnoAC            -0.0511     0.0323   -1.58   0.1137
## Garage             0.0342     0.0204    1.67   0.0948
## PoolPool           0.1180     0.0429    2.75   0.0061
## QualityLow        -0.4304     0.0593   -7.26  1.5e-12
## QualityMedium     -0.3468     0.0429   -8.09  4.5e-15
## Style2            -0.0921     0.0373   -2.47   0.0138
## Style3            -0.0228     0.0356   -0.64   0.5215
## Style4             0.0924     0.0740    1.25   0.2123
## Style5            -0.1005     0.0610   -1.65   0.0999
## Style6            -0.0369     0.0615   -0.60   0.5490
## Style7            -0.1679     0.0354   -4.74  2.8e-06
## Style9            -0.1563     0.2383   -0.66   0.5123
## Style10           -0.3320     0.2420   -1.37   0.1708
## Style11           -0.5361     0.2368   -2.26   0.0240
## HighwaynoHighway   0.1451     0.0728    1.99   0.0469
## logArea            0.7598     0.0529   14.37  < 2e-16
## logLot             0.1185     0.0190    6.24  9.2e-10
## logAge            -0.1438     0.0169   -8.50  < 2e-16
##
## Residual standard error: 0.235 on 501 degrees of freedom
## Multiple R-squared:  0.862,  Adjusted R-squared:  0.857
## F-statistic:  165 on 19 and 501 DF,  p-value: <2e-16
```

Here is some example statements:

- Multiplicative increase in price when you add one bathroom: `r 2^coefvec["Bath"]` (because I used base 2 when logging), which is 1.054 (95% CI `r 2^confintmat["Bath", ]`, which is 1.029 to 1.081).
- Doubling the area corresponds to a multiplicative increase of `r 2^coefvec["logArea"]` (which is 1.693) in price (95% CI of `r 2^confintmat["logArea", ]`, which is 1.58 to 1.82).

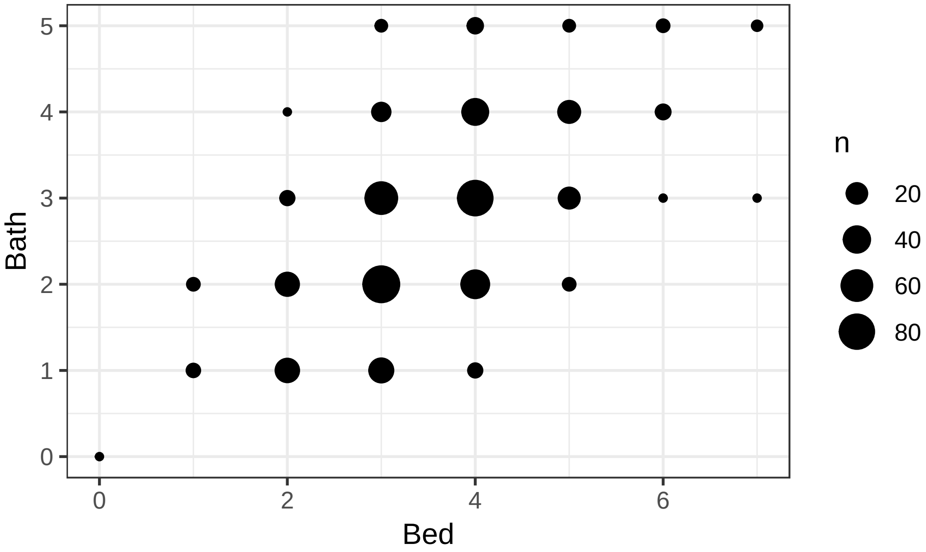
I used causal language here, this is a little loose and relaxed. In formal write ups you should use non-causal language like:

- Houses with one more bathroom tend to cost 5% more (95% confidence of 3% to 8%)

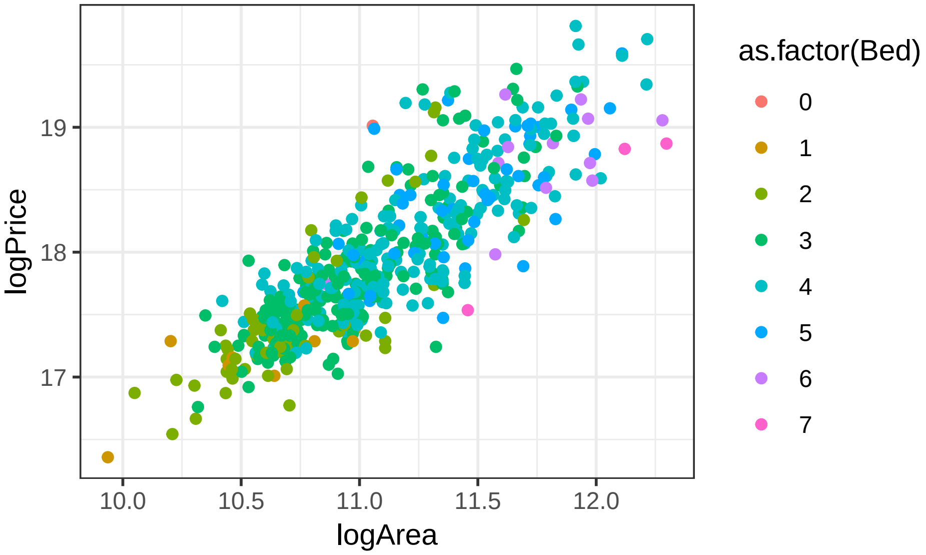
- Homes with twice the area tend to cost 69% more (95% confidence of 58% to 82%).

Interestingly, bed was not informative given the other variables. Why? Probably because it is so highly correlated with the other variables that it doesn't add any additional information on price (at least given this dataset).

```
ggplot(estate, aes(x = Bed, y = Bath)) +
  geom_count()
```



```
ggplot(estate, aes(x = logArea, y = logPrice, color = as.factor(Bed))) +
  geom_point()
```


logPrice
logArea
as.factor(Bed)