Broadway data analysis

Julianna Szabo

12/23/2020

Executive summary

Research question

Is there a correlation between the occupancy percentage of a show and the revenue per attendant? I will be looking for causality.

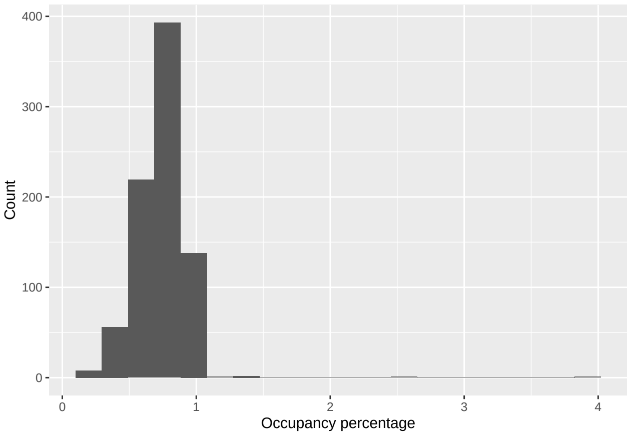
Data

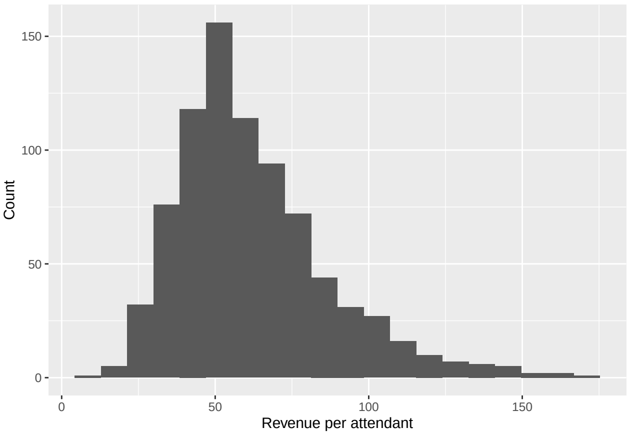
The data is very complete and representative. I have removed some missing values during the cleaning process but it was a very small percentage. Further some measures were lost by switching from a time series to a cross sectional data set. However, I aggregated on the show name, which lets me keel the most amount of detail. Most of the variables are quantitative so that means they measure what they decribe. I will use using Revenue / Attendant, where Revenue is measured as the gross revenue of the show, and attendants which are measured as total number of people who attended the show.

My x variable will be Occupancy percentage (capacity_filled) My y variable will be Revenue / Attendant which I will calculate based on revenue and attendant

There may be some measurement error in y, which is classic and doesn't affect the slope. There may be some measuement error in x which could also be classic, which does affect the slope.

Summary of variables





n	mean	median	min	max	sd
819	0.7380404	0.7463636	0.15000	3.8775	0.2051999
819	62.3321863	56.7017892	12.64152	175.1328	24.9580223

Looks like they are distributed somehwat normall, but y has a long right tail, while x has more of a left tail. Also looking at x, there are a few outliers, since a percentage should not be larger than 1. Therefore I will remove these from the set.

Ln transformations

Appendix Level- log makes the most sense

Regression Models

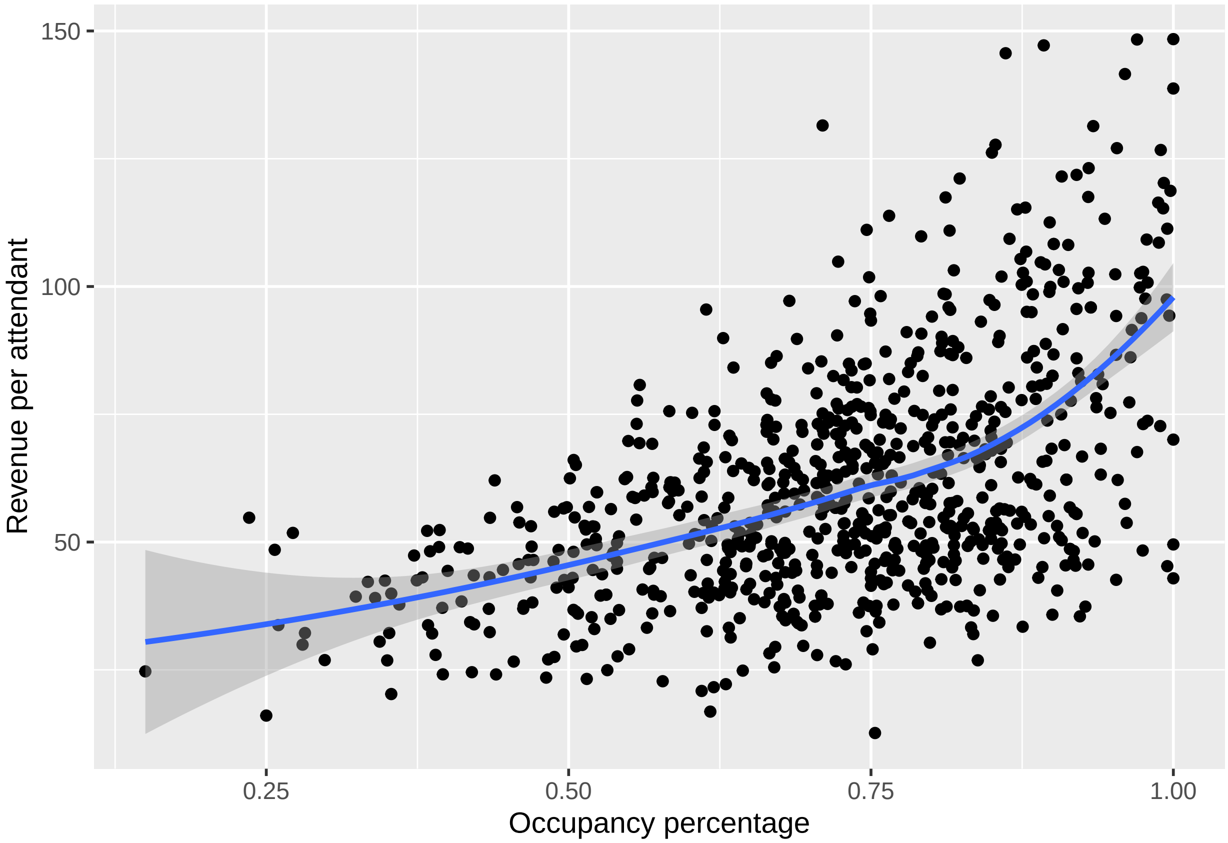
Decided on .... model

Appendix

Ln transformation

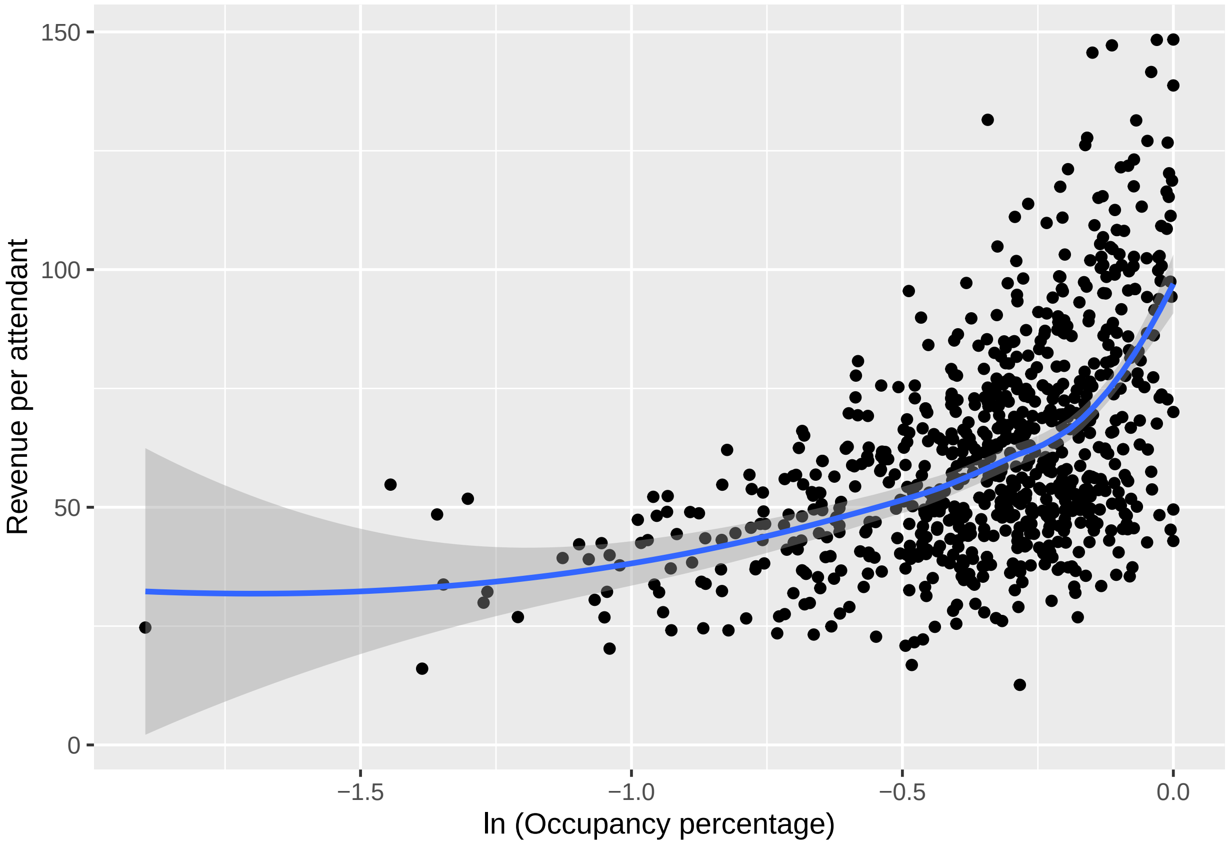
Level - level regression

```
## `geom_smooth()` using formula 'y ~ x'
```



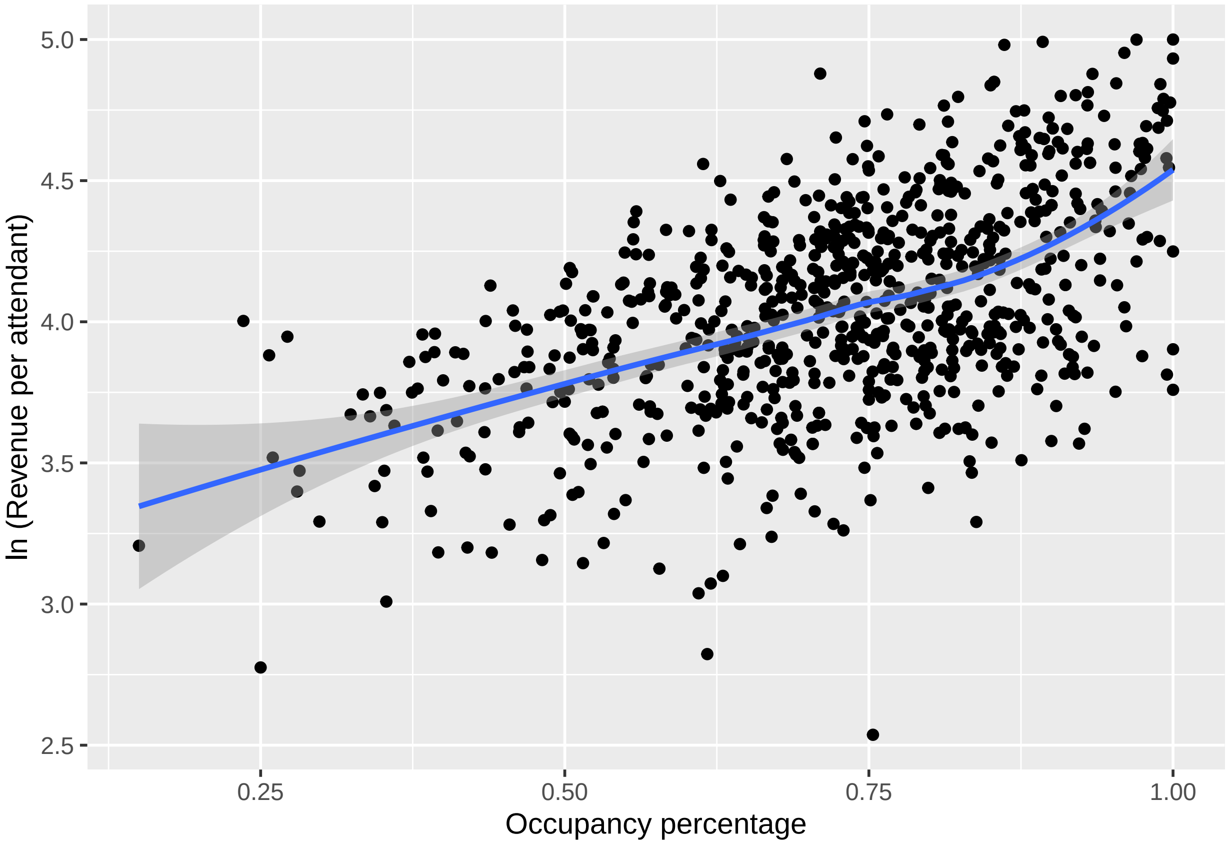
Log - level regression

```
## `geom_smooth()` using formula 'y ~ x'
```



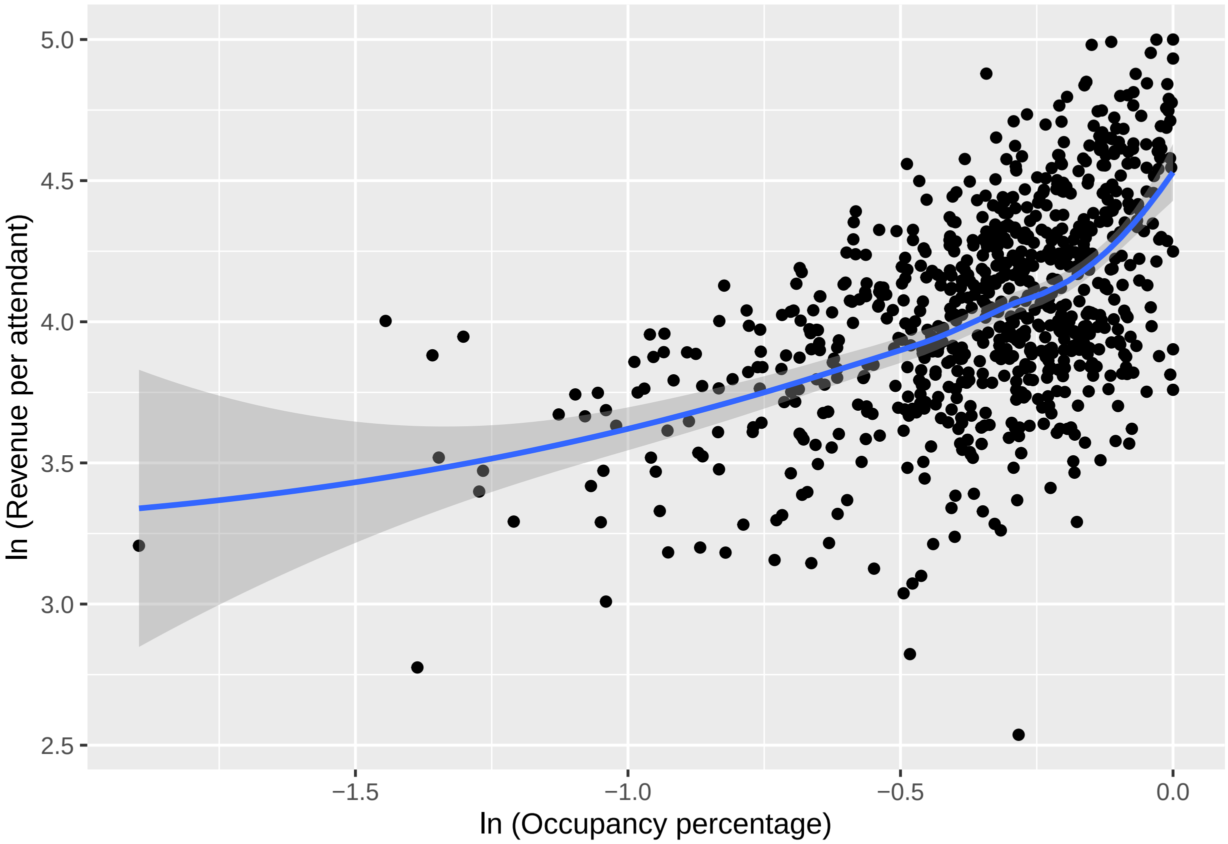
Level - log regression

```
## `geom_smooth()` using formula 'y ~ x'
```



Log - log regression

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```

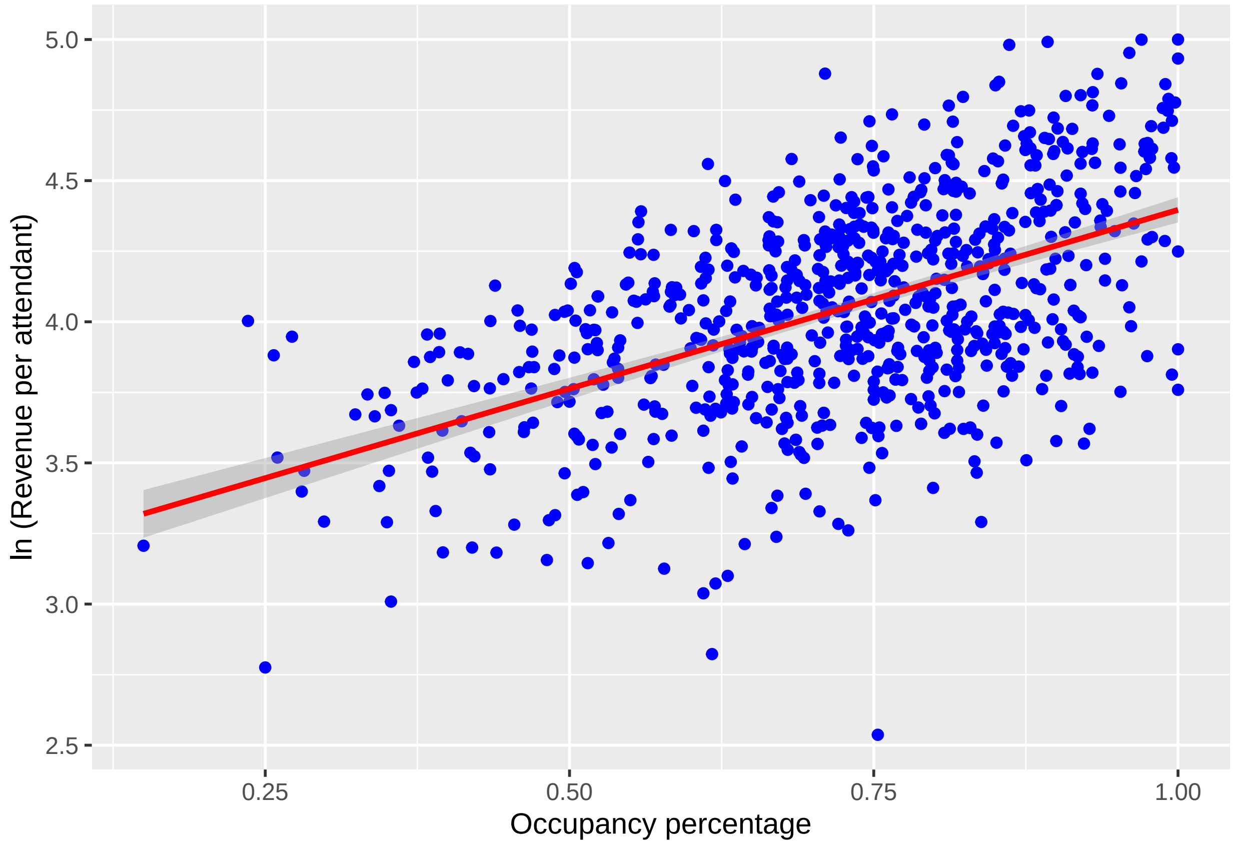


Regression modes

Regression 1 - Simple linear regression

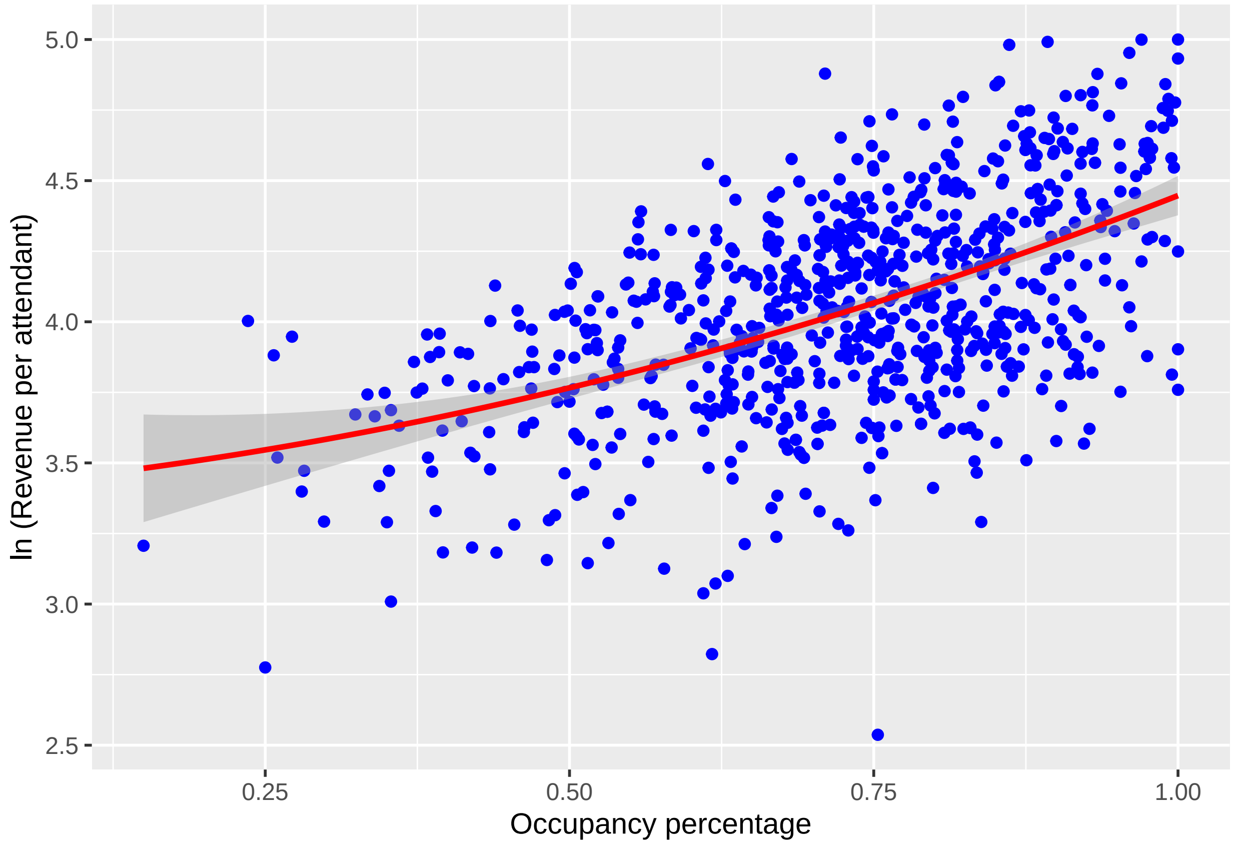
```
##
## Call:
## lm_robust(formula = ln_revenue_per_att ~ capacity_filled, data = df,
##     se_type = "HC2")
##
## Standard error type:  HC2
##
## Coefficients:
##                  Estimate Std. Error t value   Pr(>|t|) CI Lower CI Upper  DF
## (Intercept)         3.130    0.05222   59.93 1.204e-297    3.027    3.232 798
## capacity_filled     1.267    0.07224   17.54  1.769e-58    1.125    1.409 798
##
## Multiple R-squared:  0.2774 ,    Adjusted R-squared:  0.2765
## F-statistic: 307.5 on 1 and 798 DF,  p-value: < 2.2e-16
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



Regression 2 - Quadratic (linear) regression

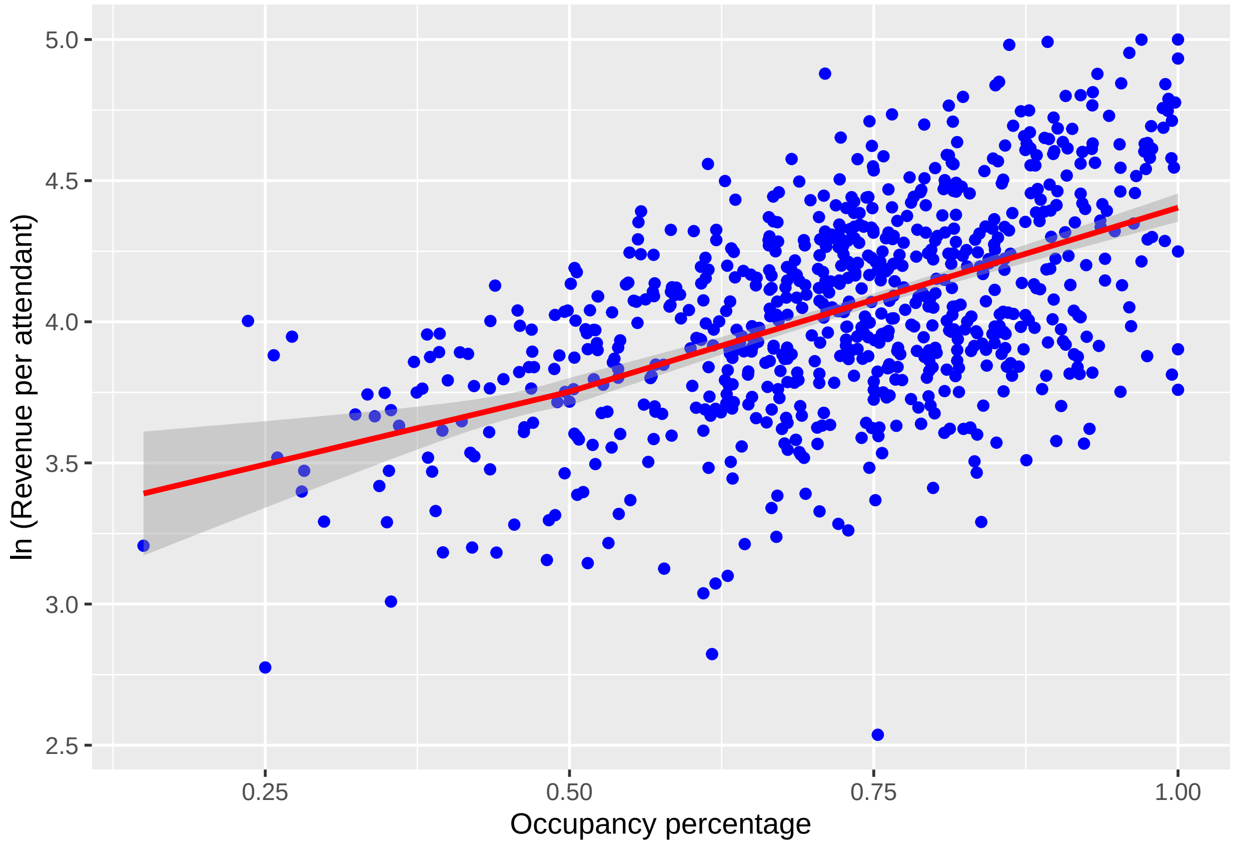
```
##
## Call:
## lm_robust(formula = ln_revenue_per_att ~ capacity_filled + capacity_filled_sq,
##     data = df)
##
## Standard error type:  HC2
##
## Coefficients:
##                      Estimate Std. Error t value  Pr(>|t|) CI Lower CI Upper  DF
## (Intercept)            3.4069     0.1663 20.4852 2.982e-75   3.0804    3.733 797
## capacity_filled        0.3969     0.5041  0.7873 4.314e-01  -0.5927    1.387 797
## capacity_filled_sq     0.6433     0.3721  1.7289 8.422e-02  -0.0871    1.374 797
##
## Multiple R-squared:  0.2805 ,    Adjusted R-squared:  0.2787
## F-statistic: 153.8 on 2 and 797 DF,  p-value: < 2.2e-16
```



Regressipn 3 - Piecewise linear spline regression

Using 0.5 as a cutof point

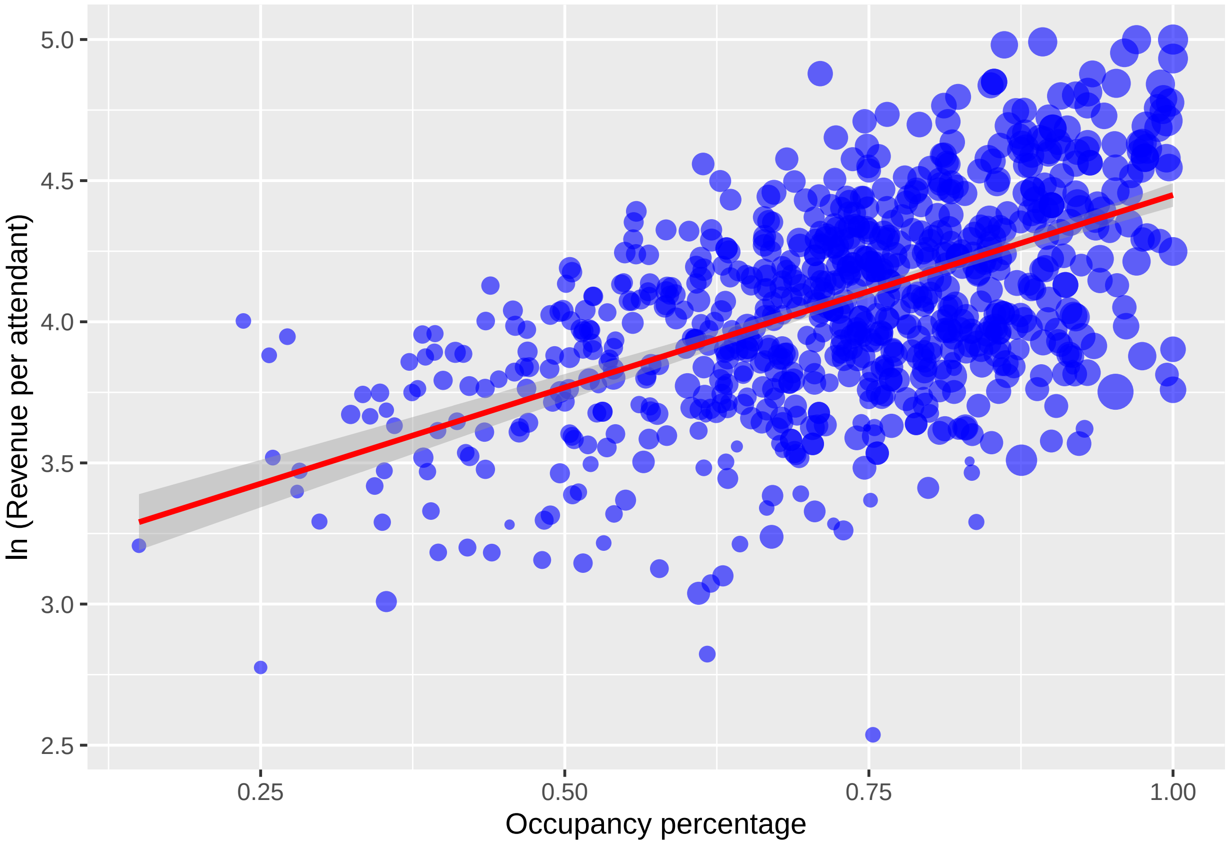
```
##
## Call:
## lm_robust(formula = ln_revenue_per_att ~ lspline(capacity_filled,
##     cutoff), data = df)
##
## Standard error type:  HC2
##
## Coefficients:
##                                   Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                          3.237    0.17047  18.989 1.293e-66
## lspline(capacity_filled, cutoff)1    1.031    0.36241   2.845 4.561e-03
## lspline(capacity_filled, cutoff)2    1.304    0.09117  14.302 1.799e-41
##                                   CI Lower CI Upper  DF
## (Intercept)                         2.9023    3.572 797
## lspline(capacity_filled, cutoff)1   0.3195    1.742 797
## lspline(capacity_filled, cutoff)2   1.1250    1.483 797
##
## Multiple R-squared:  0.2779 ,    Adjusted R-squared:  0.2761
## F-statistic: 153.1 on 2 and 797 DF,  p-value: < 2.2e-16
```



Regression 4 - Weighted linear regression, where weights = percentage of total revenue

```
##
## Call:
## lm_robust(formula = ln_revenue_per_att ~ capacity_filled, data = df,
##     weights = percentage_of_poss_profit)
##
## Weighted, Standard error type:  HC2
##
## Coefficients:
##                 Estimate Std. Error t value   Pr(>|t|) CI Lower CI Upper  DF
## (Intercept)        3.086    0.06155   50.13 8.467e-249    2.965    3.206 798
## capacity_filled    1.364    0.08552   15.94  6.827e-50    1.196    1.531 798
##
## Multiple R-squared:   0.27 , Adjusted R-squared:  0.2691
## F-statistic: 254.2 on 1 and 798 DF,  p-value: < 2.2e-16
```

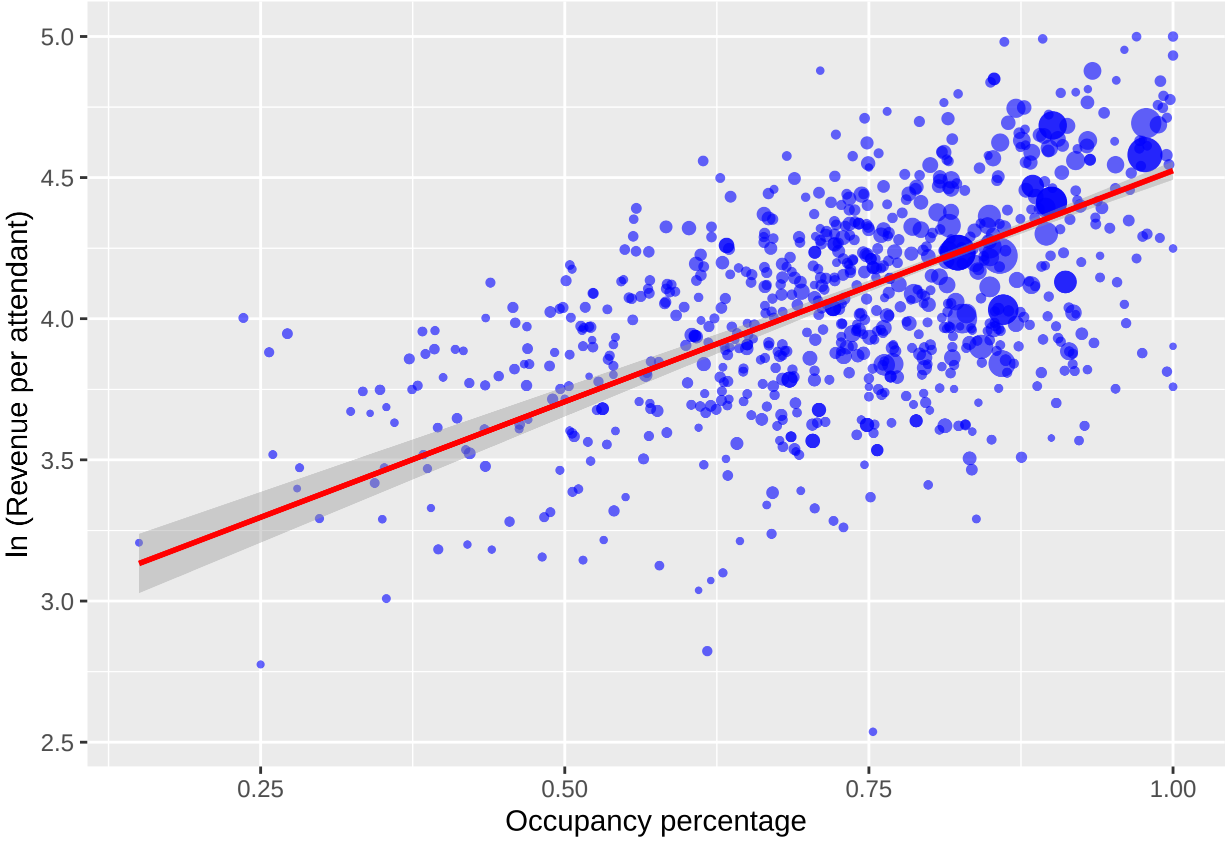
```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



Regression 5 - Weighted linear regression, where weights = number of performances

```
##
## Call:
## lm_robust(formula = ln_revenue_per_att ~ capacity_filled, data = df,
##     weights = percentage_of_poss_profit)
##
## Weighted, Standard error type:  HC2
##
## Coefficients:
##                   Estimate Std. Error t value   Pr(>|t|) CI Lower CI Upper  DF
## (Intercept)          3.086    0.06155   50.13 8.467e-249    2.965    3.206 798
## capacity_filled      1.364    0.08552   15.94  6.827e-50    1.196    1.531 798
##
## Multiple R-squared:   0.27 , Adjusted R-squared:  0.2691
## F-statistic: 254.2 on 1 and 798 DF,  p-value: < 2.2e-16
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



Model Comparison

```
## The table was written to the file '/Users/Terez/OneDrive - Central European University/Data_Analysis_
```

Additional models

Check if it becomes better if one of the weights are included as variables

```
##                              Estimate Std. Error   t value      Pr(>|t|)
## (Intercept)                 3.2974654 0.05251182 62.794729 8.277772e-311
## capacity_filled             0.4541116 0.13818230  3.286323  1.059449e-03
## percentage_of_poss_profit 0.8029153 0.13674243  5.871735  6.327487e-09
##                              CI Lower  CI Upper  DF
## (Intercept)                 3.1943876 3.4005432 797
## capacity_filled             0.1828674 0.7253558 797
## percentage_of_poss_profit 0.5344974 1.0713331 797
```

```
##                           Estimate  Std. Error   t value      Pr(>|t|)
## (Intercept)         3.146160e+00 5.31544e-02 59.189079 5.962882e-294
## capacity_filled     1.231951e+00 7.51325e-02 16.397051  2.758387e-52
## num_of_performances 2.524265e-05 8.16892e-06  3.090084  2.070540e-03
##                           CI Lower     CI Upper  DF
## (Intercept)         3.041821e+00 3.250499e+00 797
## capacity_filled     1.084470e+00 1.379432e+00 797
## num_of_performances 9.207510e-06 4.127779e-05 797
```

```
##                               Estimate   Std. Error   t value      Pr(>|t|)
## (Intercept)               3.305862e+00 5.267845e-02 62.755496 2.016309e-310
## capacity_filled           4.410309e-01 1.359712e-01  3.243561  1.229678e-03
## percentage_of_poss_profit 7.948636e-01 1.371687e-01  5.794790  9.853049e-09
## num_of_performances       1.537681e-05 6.579297e-06  2.337150  1.967842e-02
##                               CI Lower     CI Upper  DF
## (Intercept)               3.202457e+00 3.409267e+00 796
## capacity_filled           1.741264e-01 7.079354e-01 796
## percentage_of_poss_profit 5.256085e-01 1.064119e+00 796
## num_of_performances       2.461984e-06 2.829163e-05 796
```

Explore again

```
## The table was written to the file ’/Users/Terez/OneDrive - Central European University/Data_Analysis_
```