Visualisation: Modelling the World

Richie Morrisroe

July 14, 2019

What is Visualisation?

- a tool for understanding the world
- a way to communicate a particular perspective on data
- an adjunct to thought

Why Visualisation?

- The eye is really really good at finding patterns in pictures
- in fact, it's so good that it can find patterns that aren't even there

Figure: What do you see?

The importance of perspective

- You can see one of two things in the previous image
- Which of them can depend on what you expect to see
- It can also depend on what your environment contains

Muller-Lyer

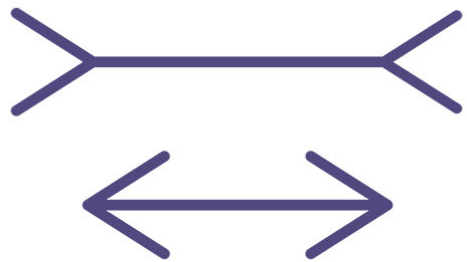


Figure: Which line is longer?

This illusion doesn't affect everyone similarly

- Europeans and Americans are more susceptible
- Africans are less susceptible
- Possibility that due to presence of right angles in urban environments
- appears to be a small difference between urban and rural dwellers

Who cares?

- Shows that how we interpret stimuli is not tabula rasa
- When you gaze into the image, the image also gazes into you...
- We bring our own perception and previous associations into any image [1]

[1] anything really, but we're talking about images here.

When to use Visualisation?

{Always}

Running Example

- Property Price Register
  - Kinda a crappy dataset
  - No cleaning or checking done by the authority
  - lots of craziness (1 apartment for 18.6mn)

Property Price Register

- We used Google's geocoding service to get more details on each observation
- I updated Shane Lynn's script and ran it on the data up till October 2018
- I also typically break out properties sold for greater than 1e6, as they are often multiple-unit sales (and there's little to no automated way of figuring this out) [2]
- Lots of manual fixing required
- the irish text definitely doesn't help

[2] please someone in the audience suggest a better idea

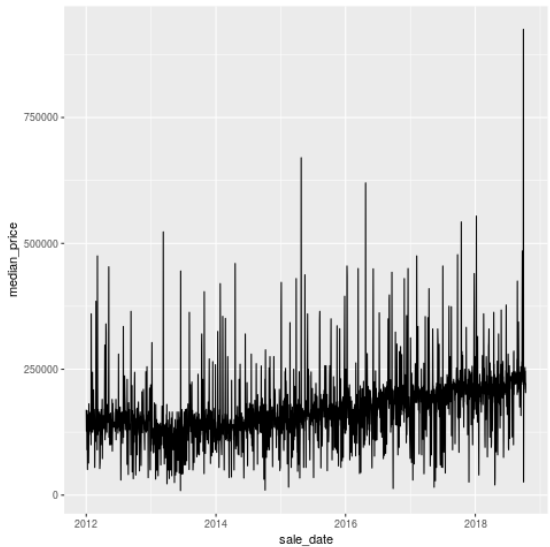
Assumptions of Statistical Graphics

- there are many
- in this section, I'd like to subvert them, in order to make you think

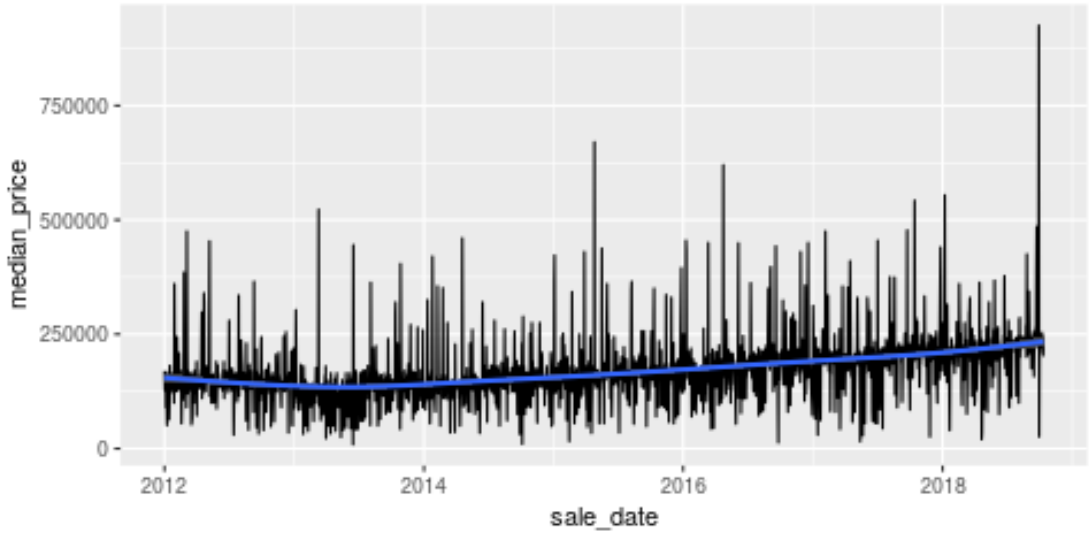
Line Graphs

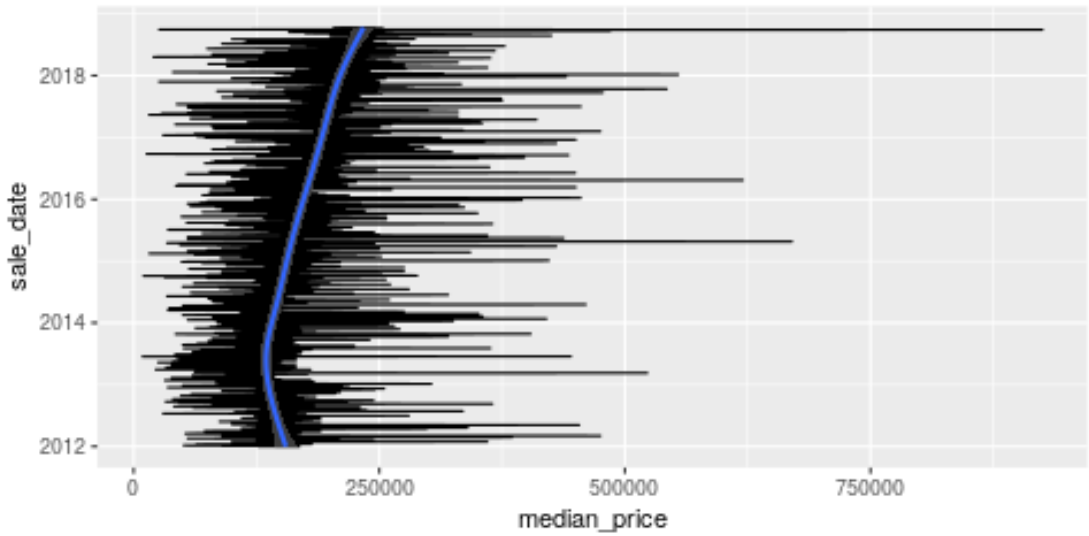
- Normally represent time
- scatterplots don't (always) have the same assumptions
- what is the deepest assumption?

Median Property Price by Day, Ireland 2011-18



Flipped Line Chart





F-ing Line Chart

```
ggplot(median_price_by_day, aes(y=sale_date, x=median_price)
```

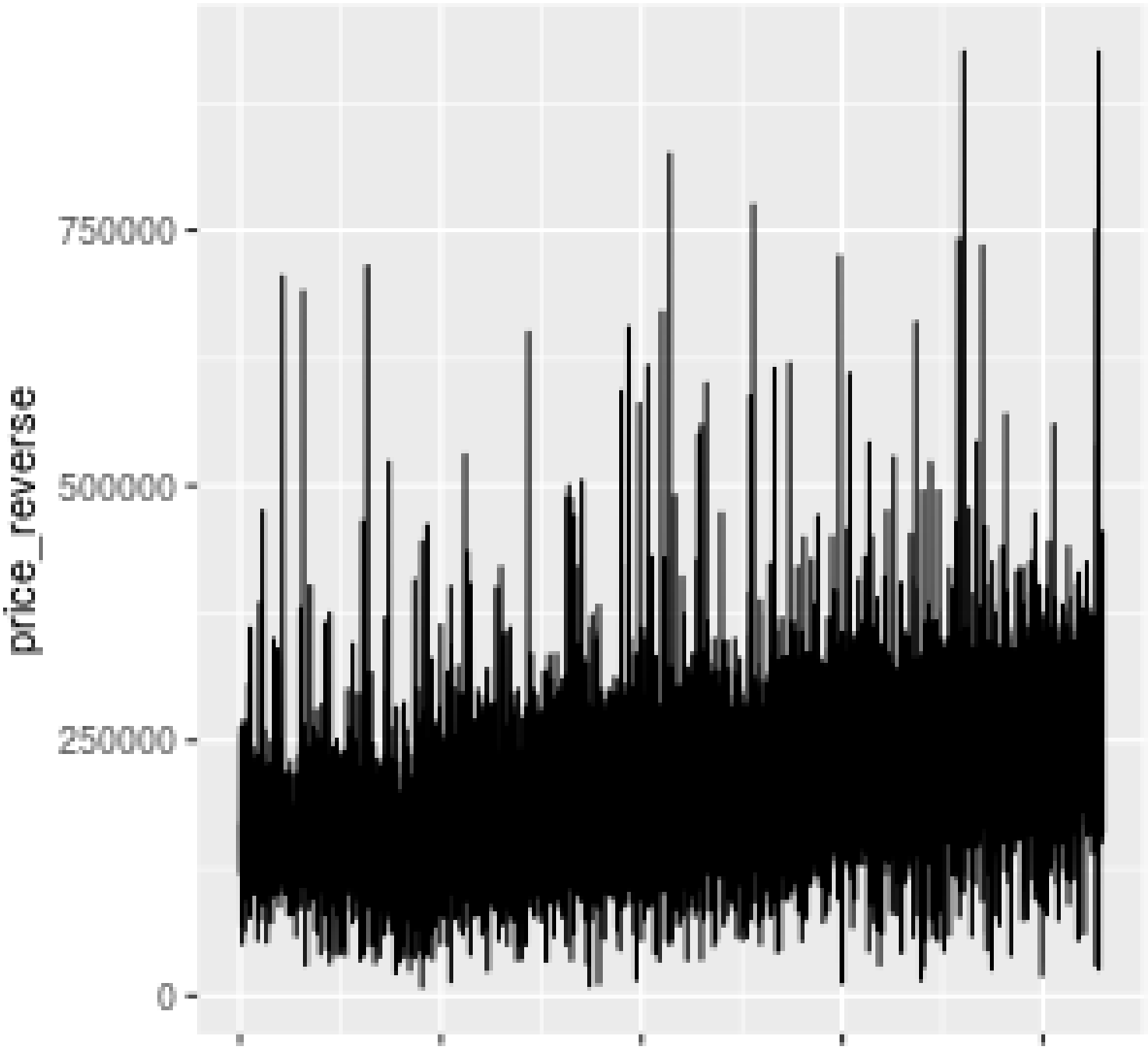
#+attr$_{latex}$:width 150px :height 150px

- Here, the violence is that we swap the axes in a fashion only a monster would

Abusing Standard Assumptions

```
ggplot(median_price_by_day, aes(y=sale_date, x=median_price)
```

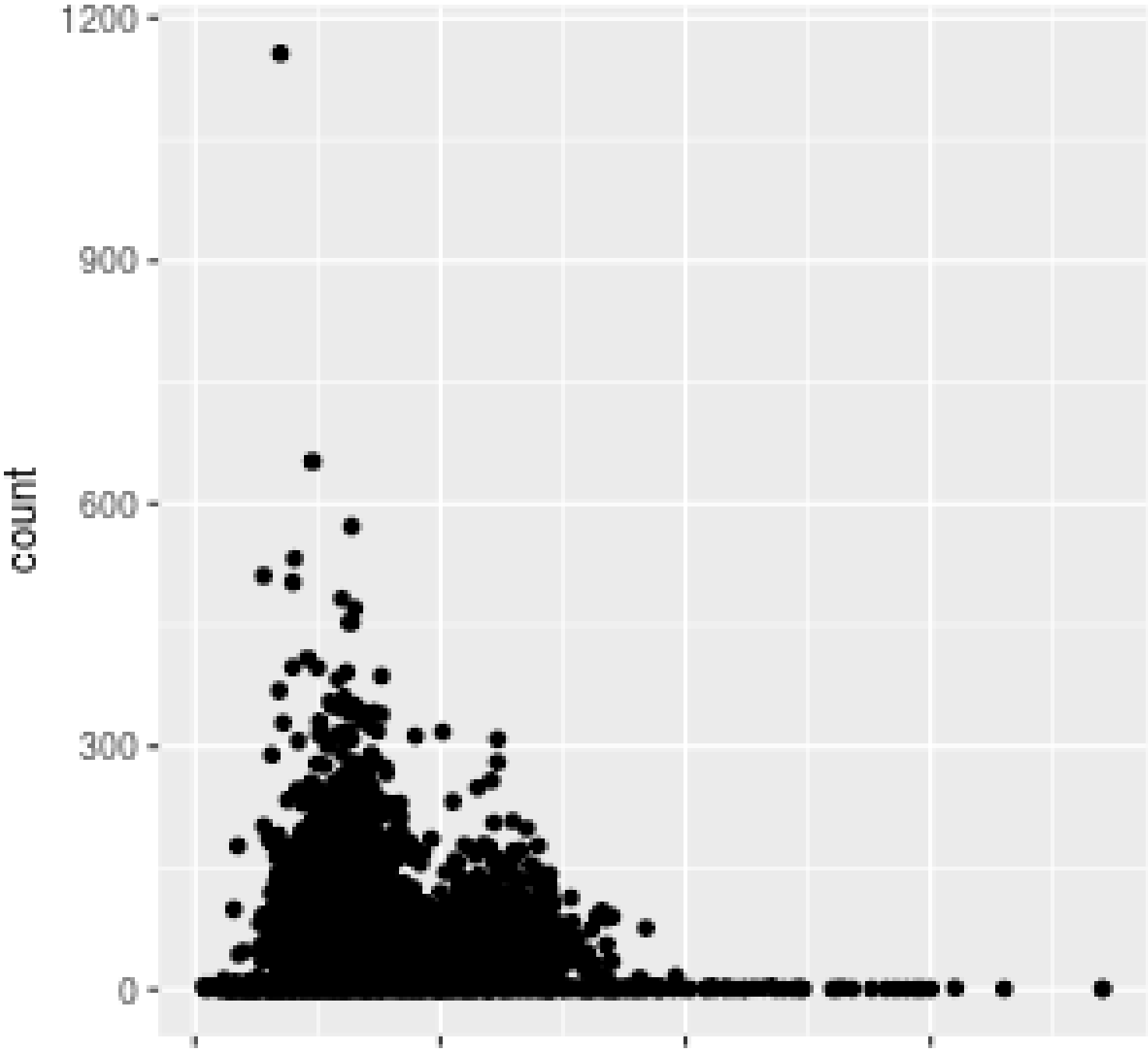
Backwards Line Chart



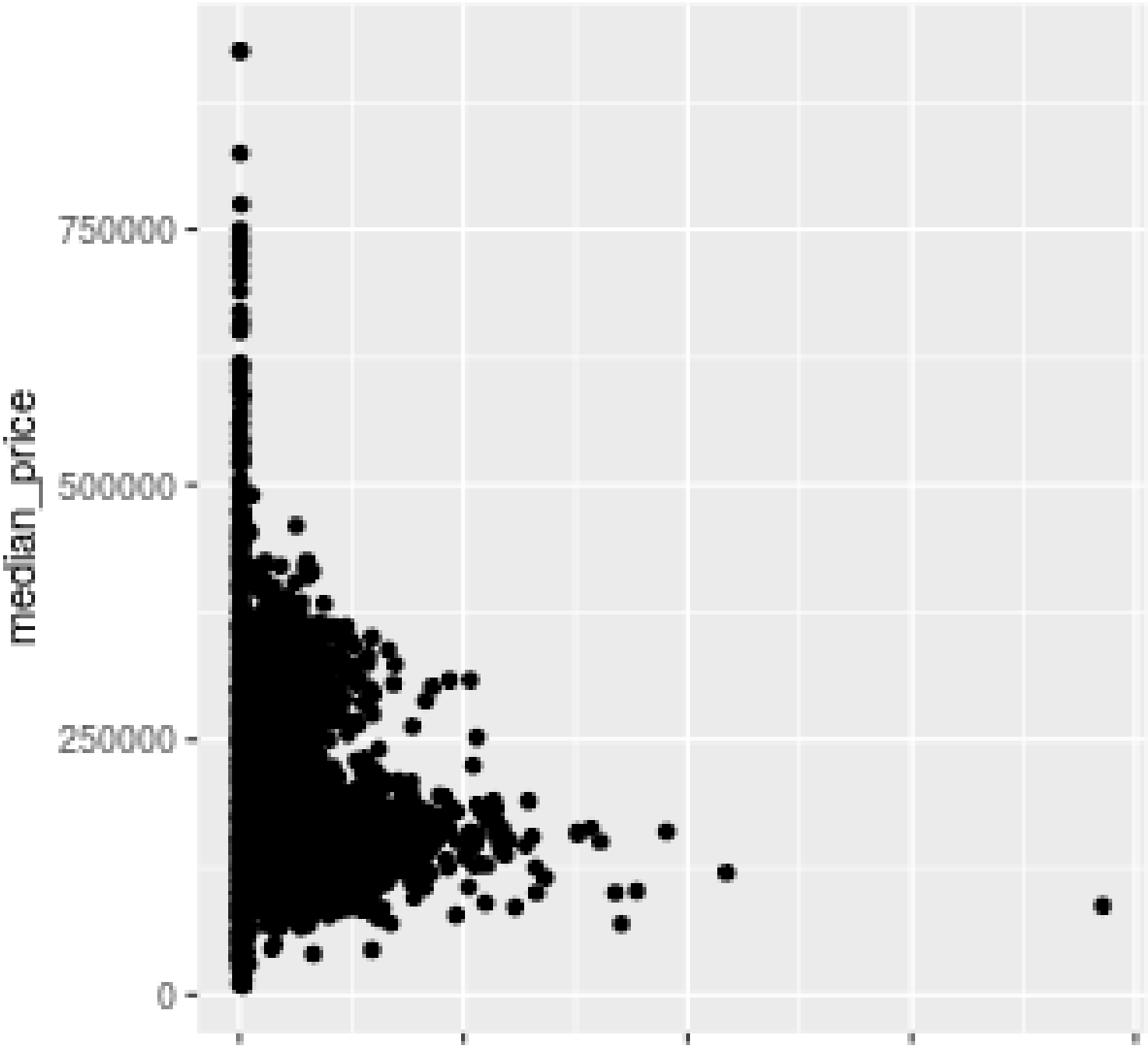
Scatter plot

- Also encodes a set of base assumptions
- points nearer to each other in space are more related
- more orientation issues

Standard Scatter



Flipped Scatter



What does this tell us?

- We have a base level of assumptions that we bring to graphics (especially statistical graphics)
- Most of these appear to have been formed by Descartes
- When these assumptions are subverted, expect problems

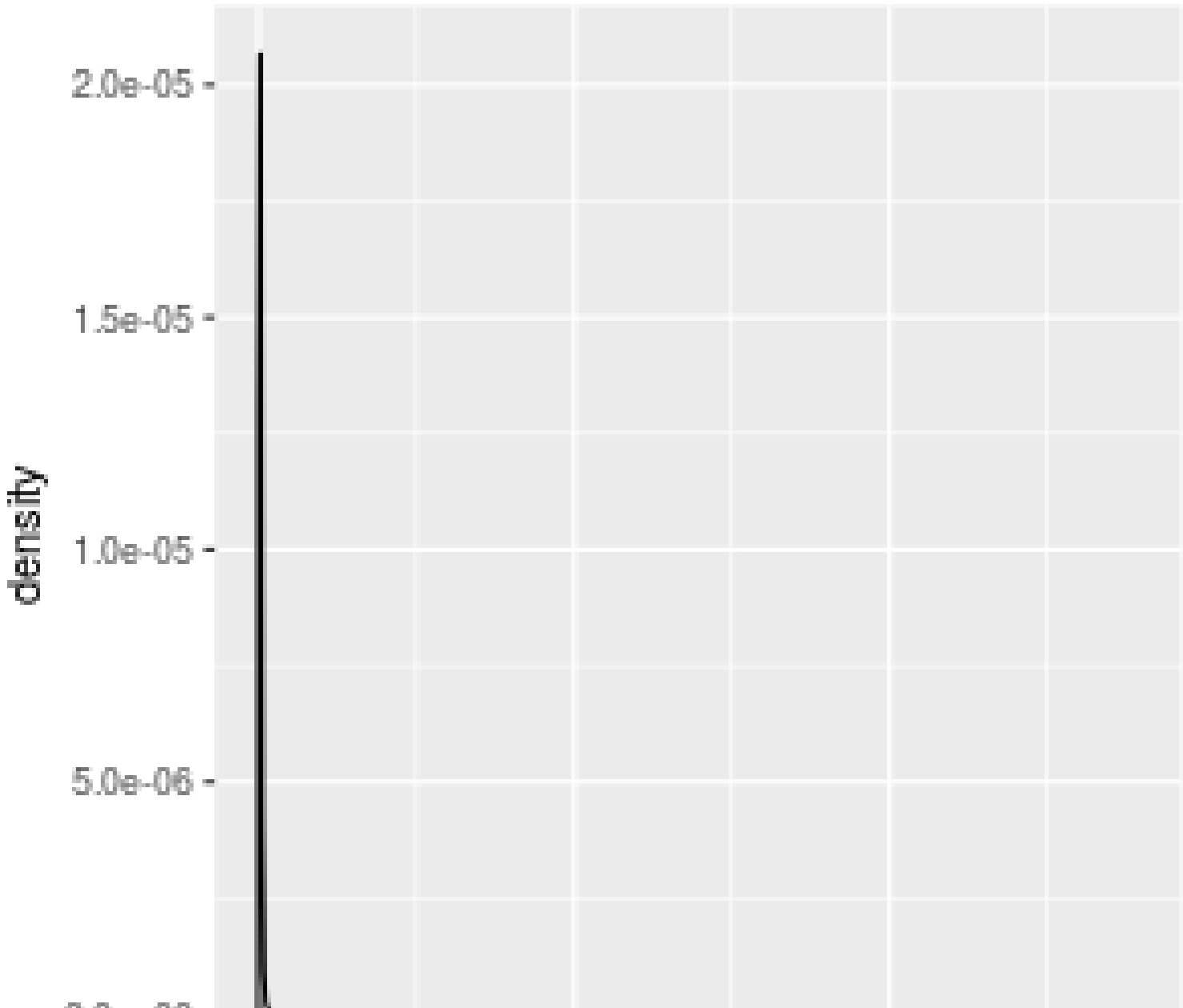
Simple Statistical Graphics

- Graphs excel at showing relations between things
- Consider the difference between quantiles of a variable, and a density plot
- For example, the price of houses:

0%	5079
10%	55000
20%	85000
30%	115000
40%	145000
50%	175000
60%	214000
70%	255505
80%	315000
90%	430000
100%	139165000

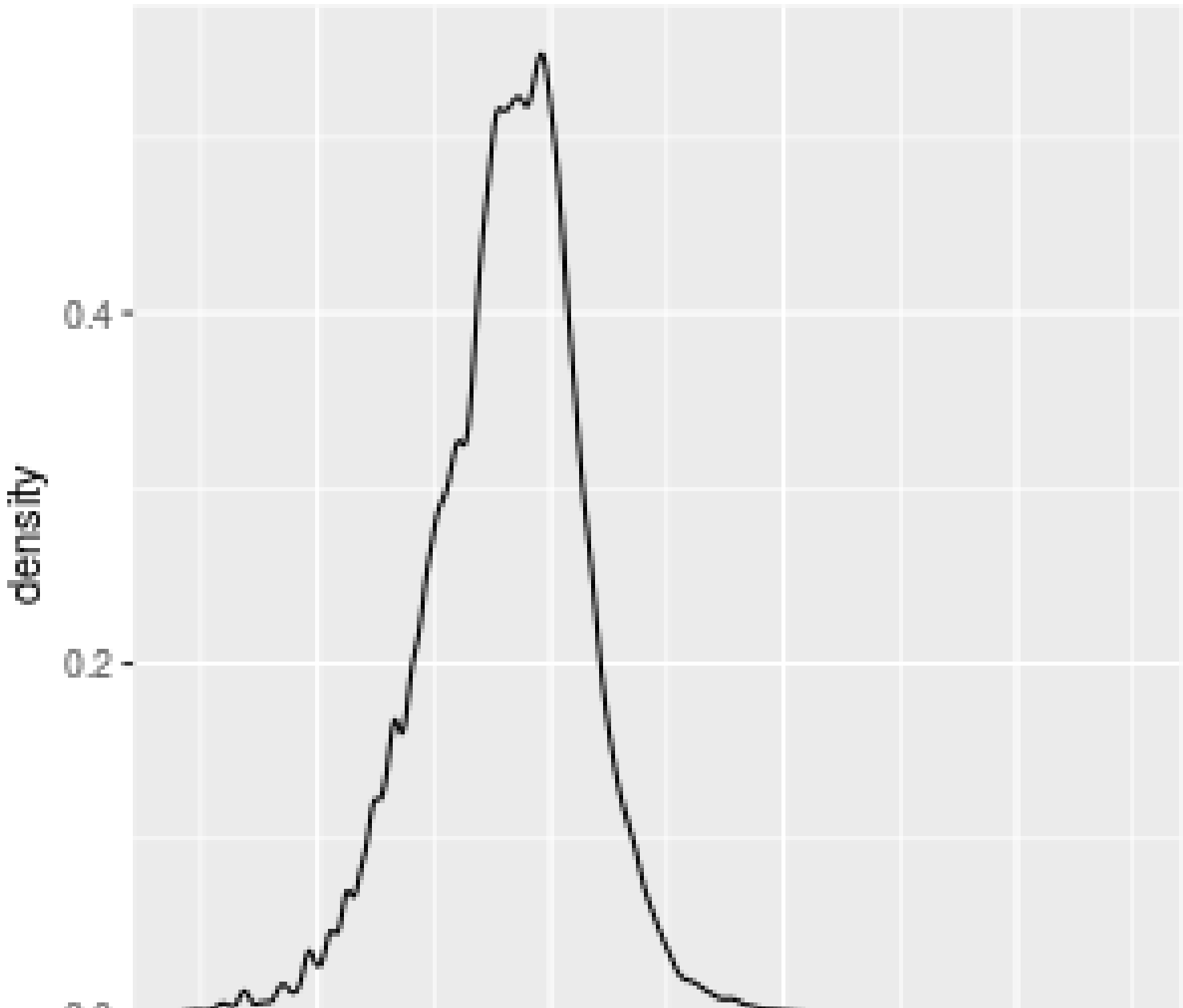
Density Plot

#+attr$_{latex}$:width 150px :height 150px



Better Density Plot

#+attr$_{latex}$:width 150px :height 150px



Transformations

- Useful to get a better sense of the data
- Have a bunch of assumptions (what's the log of -1)
- Can be used to deceive very, very easily
- Really really useful in everyday practice

Getting the sense of things

- Picking the right visualisation for the data is important

```
ggplot(dubcity, aes(x=sale_date, y=price))+geom_point()
```

- is this a good plot?
- does this depend on the number of points?

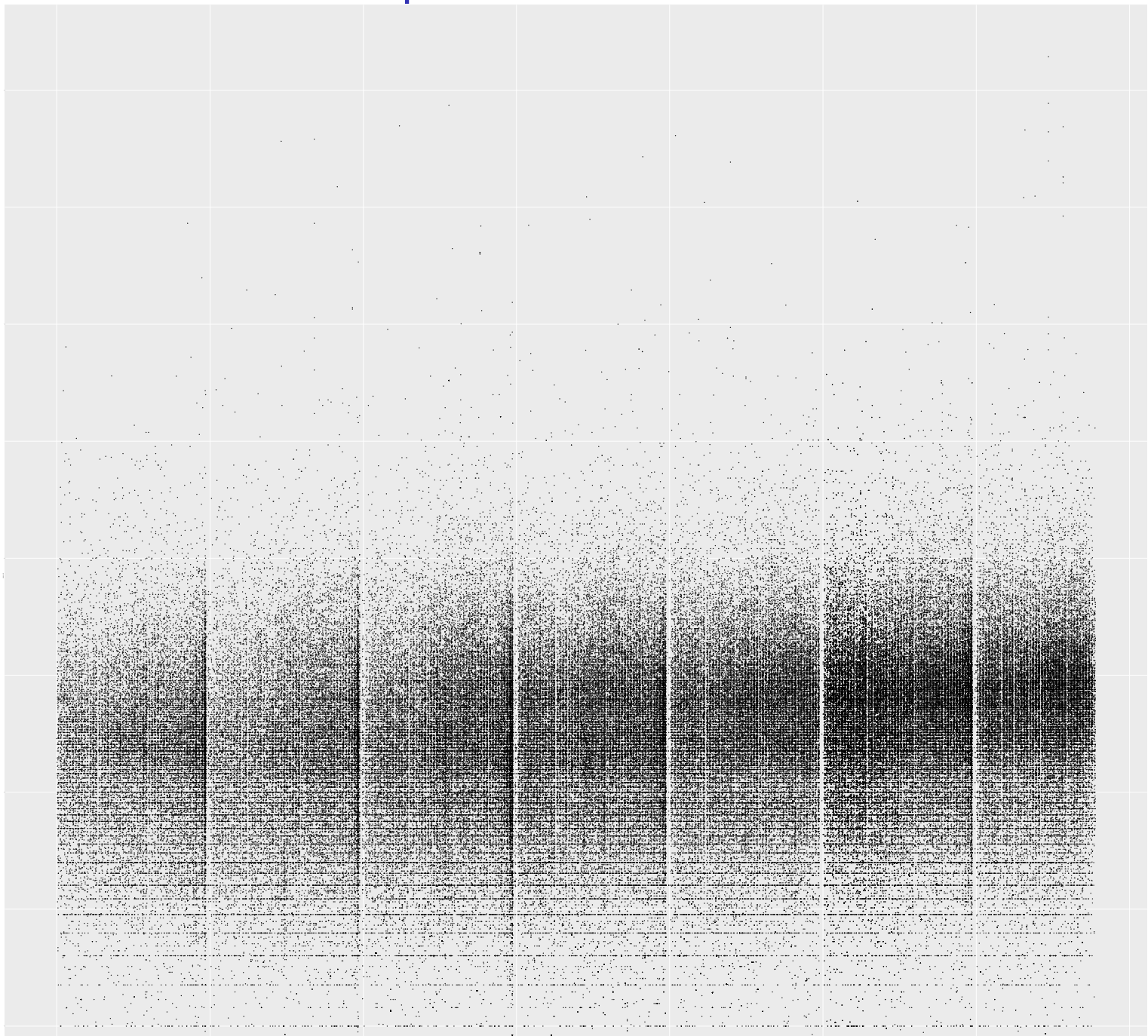
Sampling and Plotting

```
ggplot(dubcity_samp, aes(x=sale_date, y=price))+geom_point()
```

#+attr$_{latex}$:width 150px :height 150px

- Not really

Transformations Help



No data is an island

- The first obvious thing is to split by county, right?

```
ggplot(ppr_gc_smaller, aes(x=sale_date, y=log(price, 10)))+g
```

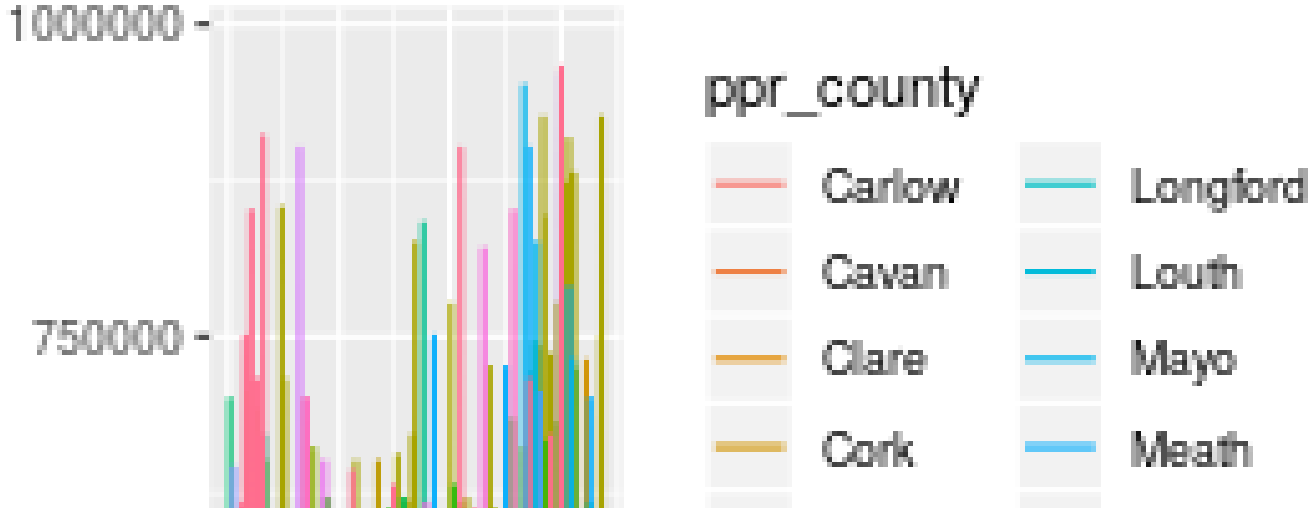
#+attr$_{latex}$:width 150px :height 150px- Oh look, it's lot of little boxes of crap :(

Summarisation

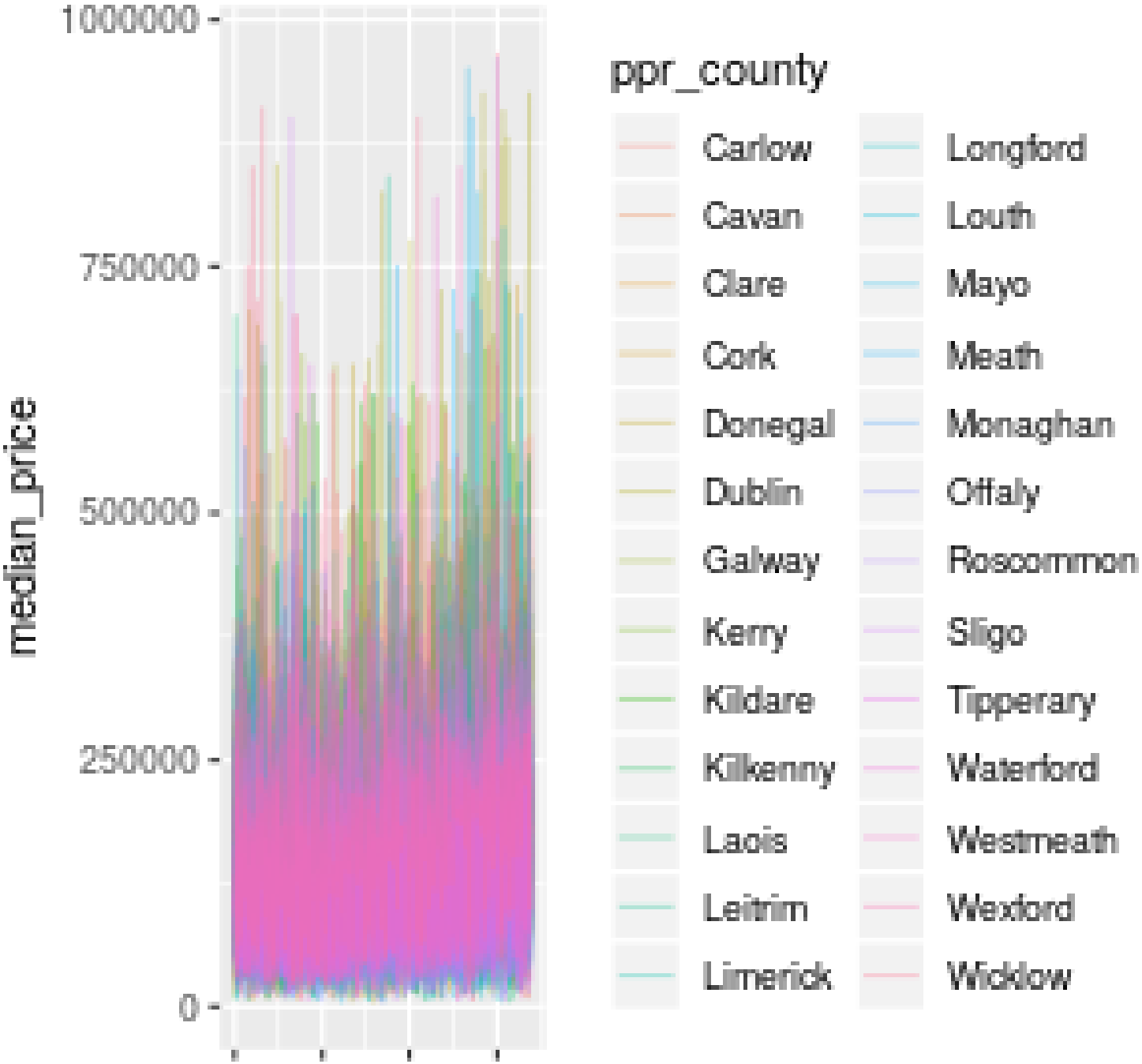
- The obvious answer is summarisation

```
county_daily <- ppr_gc2 %>% group_by(sale_date, ppr_county,
  summarise(count=n(), min_price=min(price),
            median_price=median(price),
            max_price=max(price)) %>%
  mutate(min_to_median=min_price/median_price,
         max_to_median=max_price/median_price,
         max_to_min=max_price/min_price)
ggplot(county_daily, aes(x=sale_date, y=median_price, colou
```

#+attr$_{latex}$:width 150px :height 150px



Reducing Alpha kinda works...



A redundant faceting variable

- We just group by a higher level variable

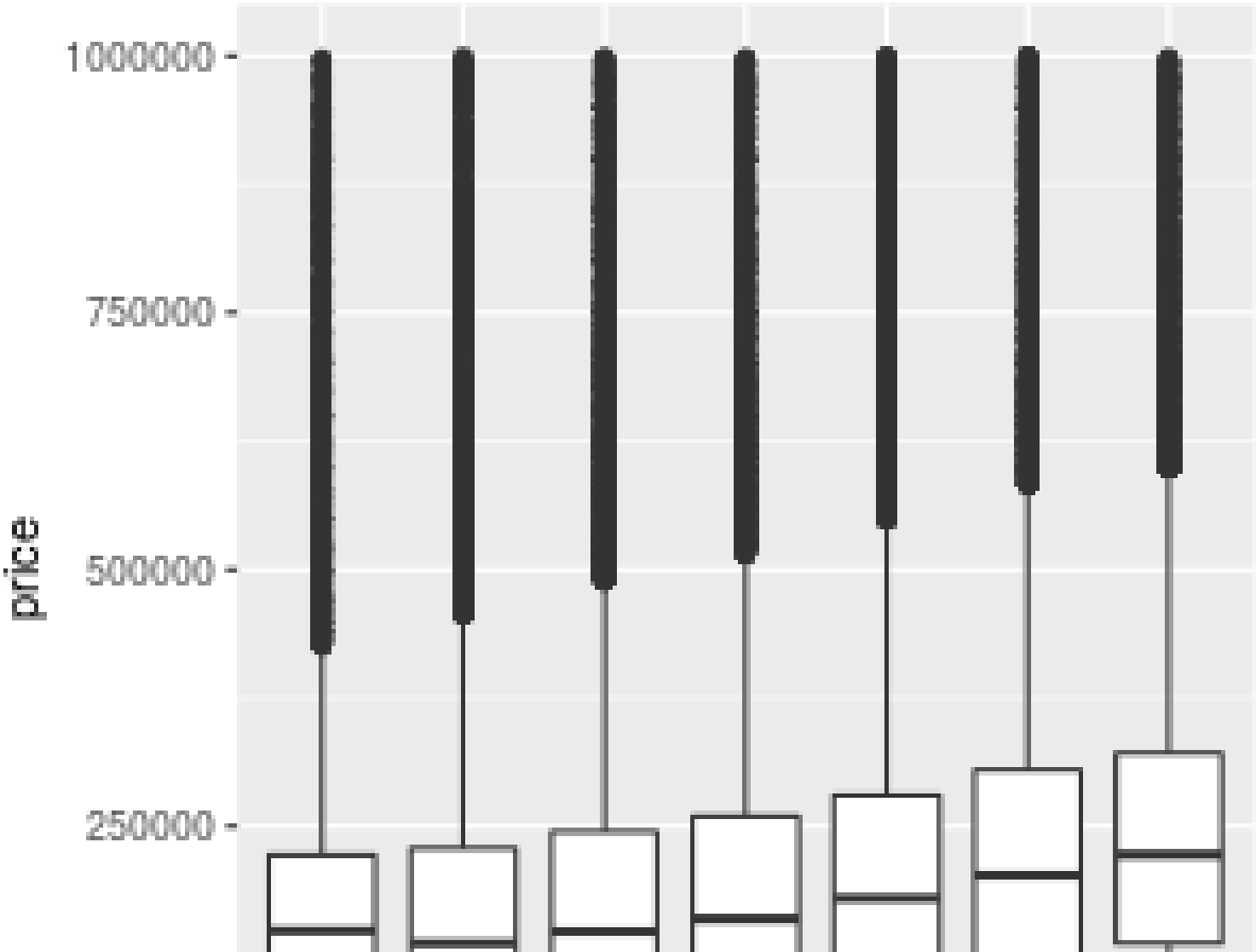
WTF?

- This is one of the major advantages of visualisation:
  - it helps to (dis)confirm your assumptions
  - given that we have too many lines in the various groupings,we know that somethng has gone horribly wrong
  - in this case, it's a mismatch between two different types of data

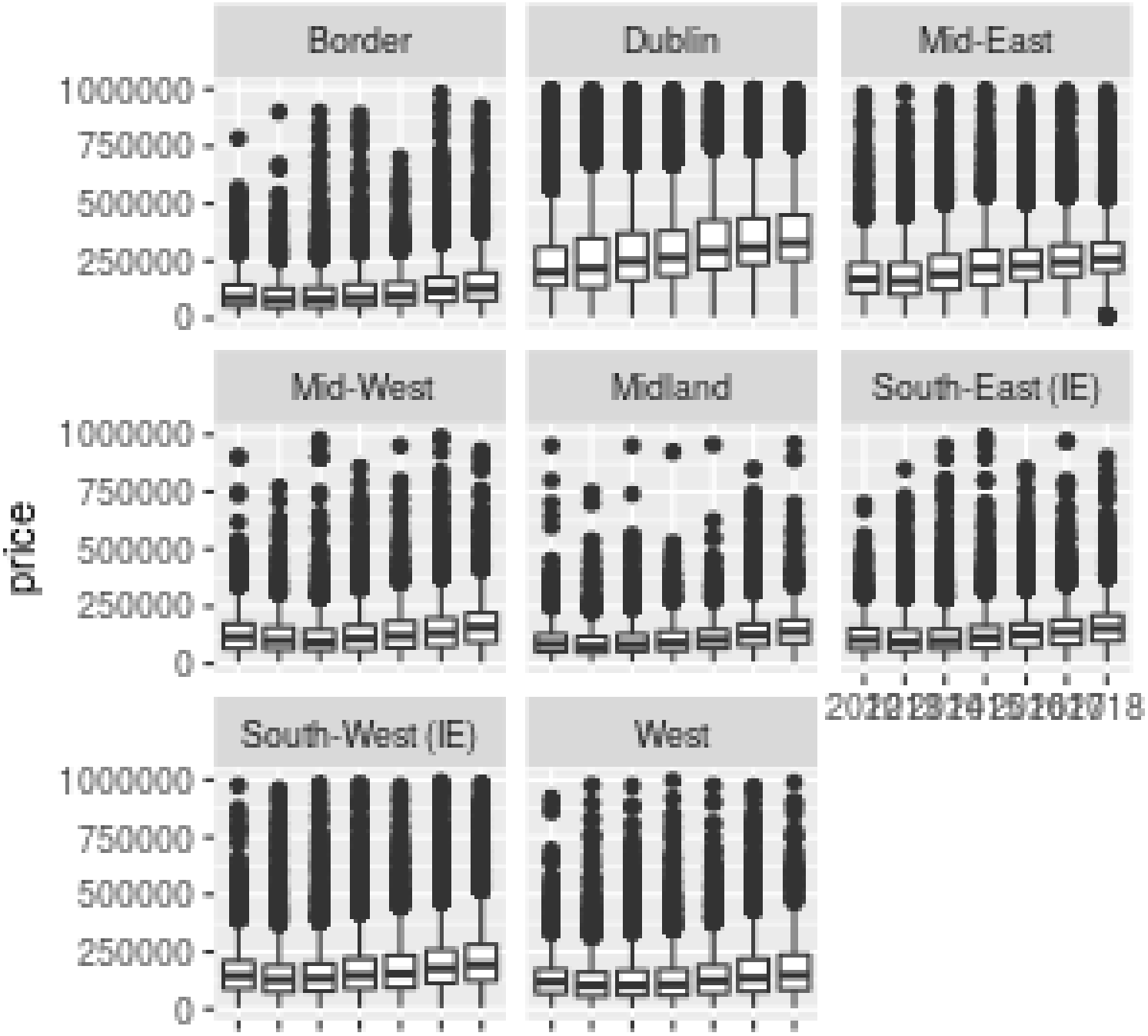
Distributions (i.e. boxplots)

```
ggplot(ppr_gc2, aes(x=as.factor(year), y=price))+geom_boxplo
```

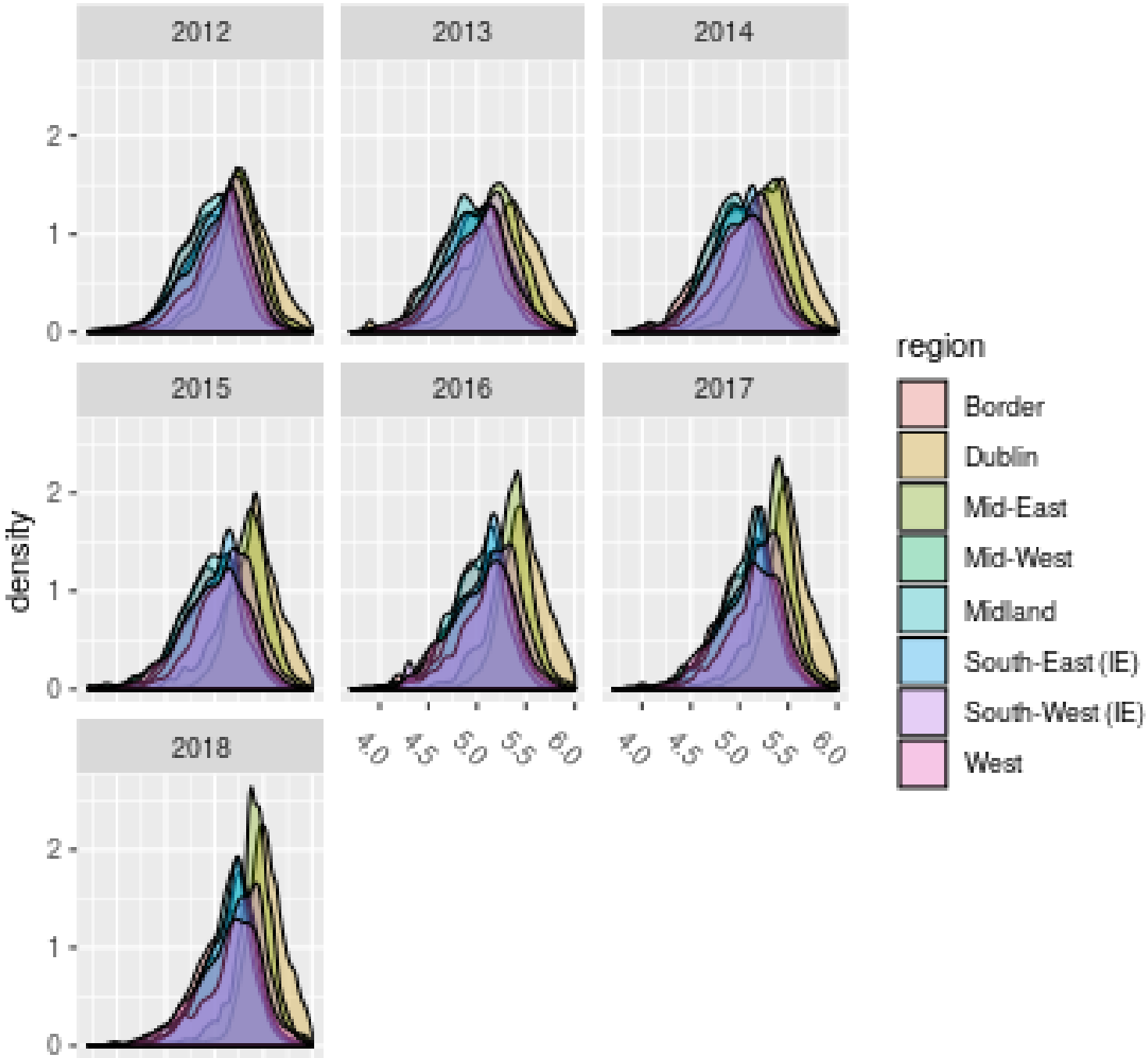
#+attr$_{latex}$:width 150px :height 150px



Faceting, redux



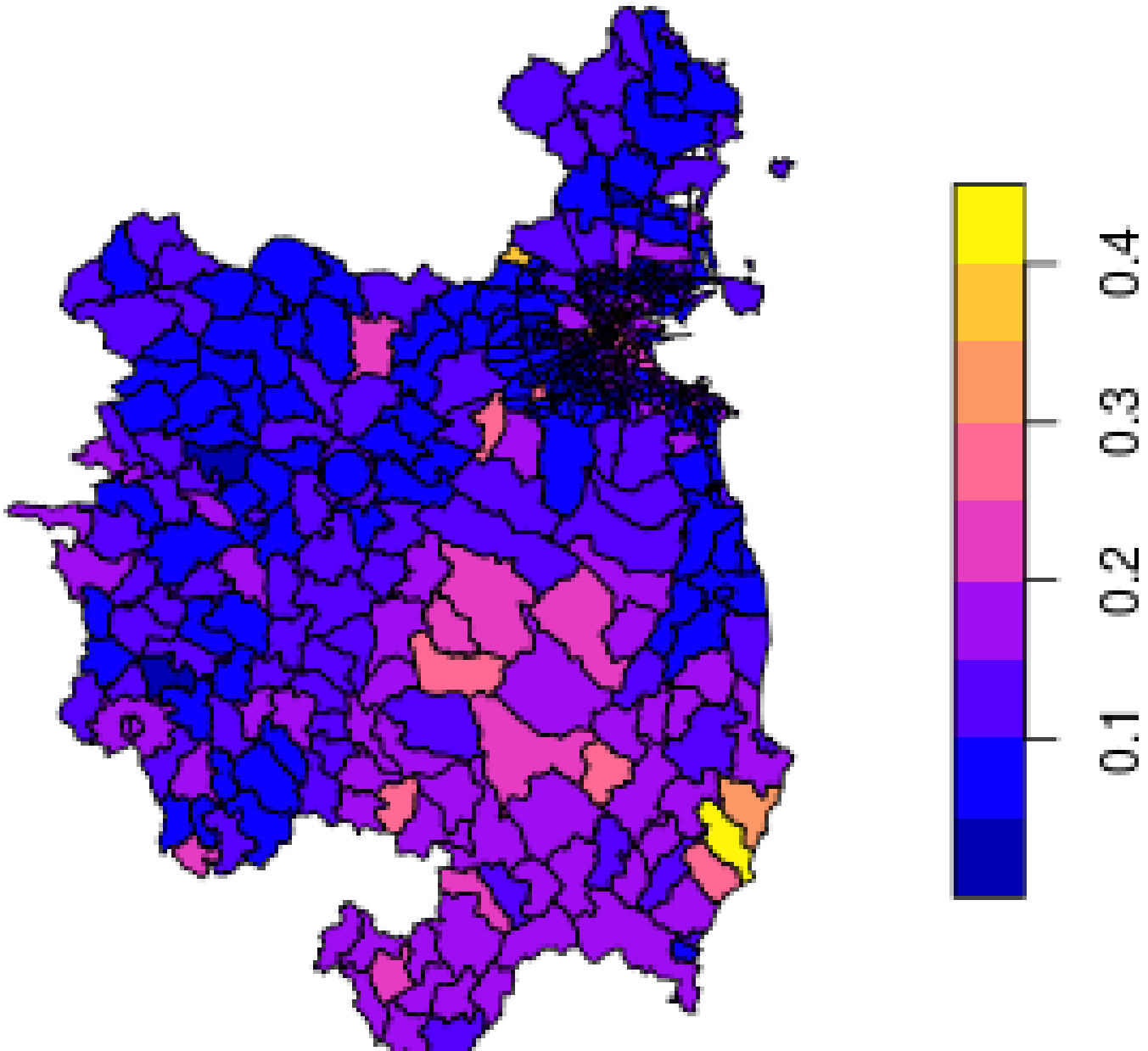
Distributions over Time, Redux



Spatial vs Temporal

- line plots vs maps
- time versus space
- both provide insight into
- pick one, difficult to do both

Line plots ignore space, maps ignore time



Performative vs Presentation

- Two types of graphs:
  - for yourself
  - for other people (and different audiences need different things)

Performative Graphics

- These are used to help you understand a problem
- typically created in an iterative fashion
- often move from data transformation to visualisation and back agai

Presentation Graphics

Different Audiences/story

- To some extent, your job with presentation visualisations is to tell a story
- hopefully, it will be nuanced, but that isn't a requirement [3]
- Often good to show smooths as opposed to raw data
- raw data is often ugly
- need for care here, as this should only be done where there is a clear effect

[3] and in fact, it may be better to remove all nuance from the presentation and provide a longer document with all the failed approaches and hacking needed to actually reproduce your results

Interactivity and Dashboards

- Can show both time and space
- for reporting, these are essential
- Much more effort from a software-engineering perspective [4]

[4] for me, at least

Reporting

- Some times you need to repeat yourself
- Couple of ways of approaching this
  - Dashboards
  - Automated Reports

Dashboards

- Lots of effort to set up correctly
- typically need a bunch of ETL to get data into correct format
- Low-maintenance once the original work is done
- Much more useful for business users

Automated Reports

- Less effort to get working (especially with Sweave, knitr and org/pandoc)
- A lot more effort to get working in a Python/SQL context
- More maintenance over time (someone needs to update the report)

Principles of Reporting Visualisations

- Time view essential
- preferably forecasts, with results of previous forecasts
- allows
- Simple, simple, simple
- One clear message (key metric or whatever)
- available material for those that want to dig deeper