DA3_1_koncz_tamas_161117

Tamas Koncz

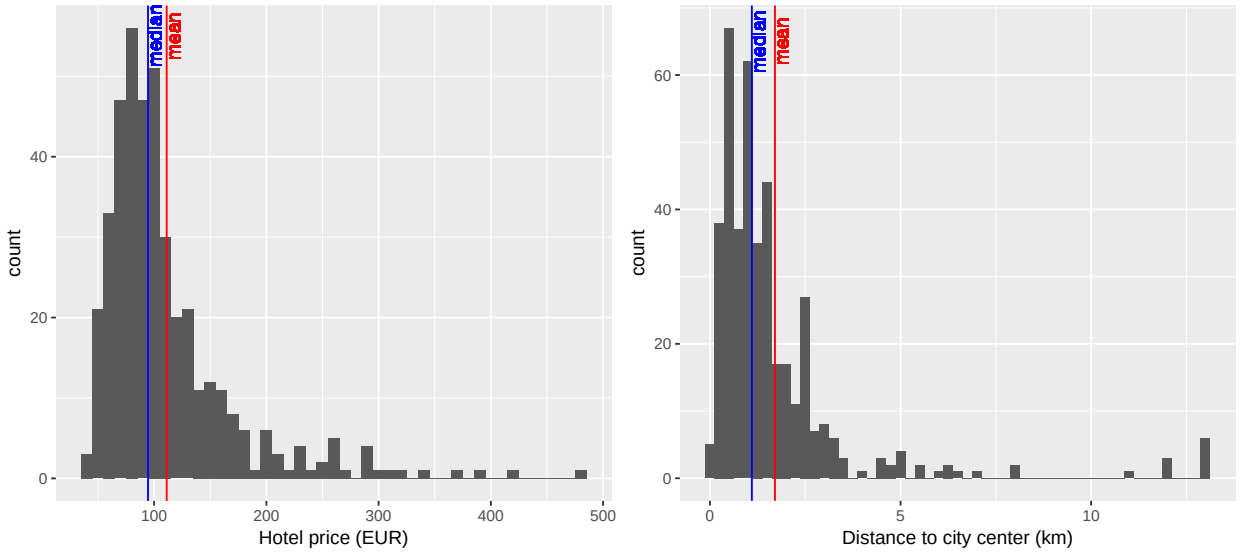
2017 november 16

1. Filter the data to the city of your choice and other characteristics (stars, accomodation type) . Describe the distribution of the price and distance variables. Comment on graphs. (1-2 sentences)

Loading data & filtering:

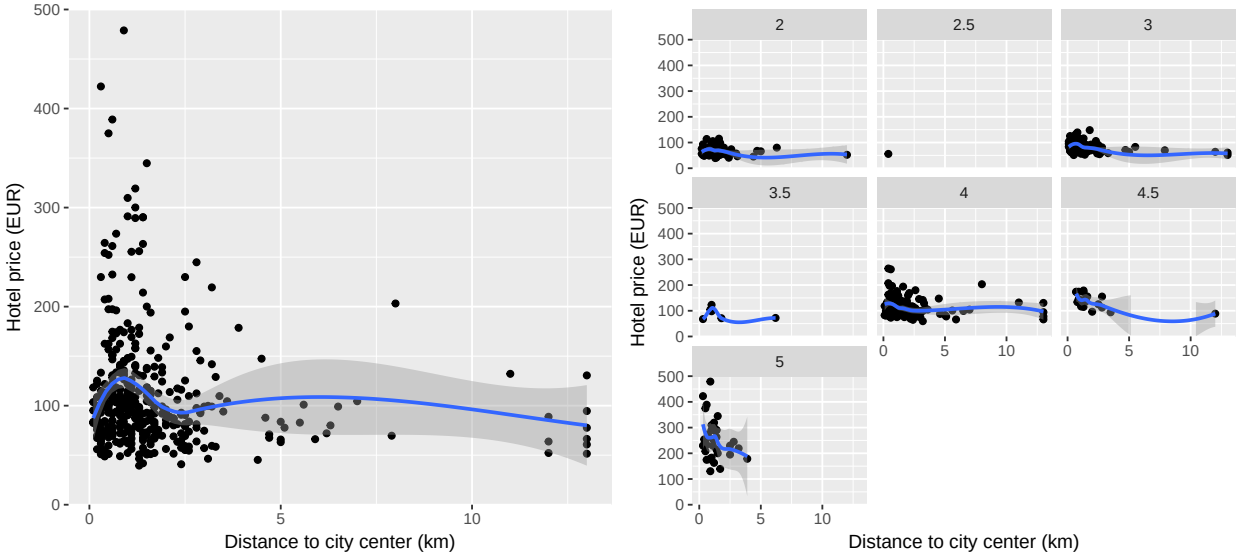
```
hotels <- fread('hotels_all_nov21.csv')

dt <- hotels[accommodation_type %in% c('Hotel', 'Hostel'), ]
dt <- dt[stars >= 2, ]
dt <- dt[city == 'Barcelona', ]
```

Let's look at the distribution of the 'price' and 'distance' variables (codes are hidden with 'echo = FALSE' for better readability of the final document):



As we can see, both are skewed to the right (means are larger than medians), with long tails. The hotel prices' distribution is very similar to what we can usually observe for price variables. Also, if we think about the accomodation business, it only make sense to have most hotels close to the city center, rather than in suburban areas.

Next we can plot the distance against the price on a scatterplot:



The relationship between the variables are not straightforward based on a scatterplot visualization. With the help of a loess smoothing, we can also observe that it is likely not linear across all distances.

The right hand chart is the same scatterplot but breaken down to different hotel star categories. We can observe significant differences in the distribution among prices in the different categories, that could be part of the explaination for non-linearities. (As I've seen no clear data problems that should have been addressed)

2. Sample definition: You may or may not want to drop some observations; make a choice and argue for it (1-2 sentences).

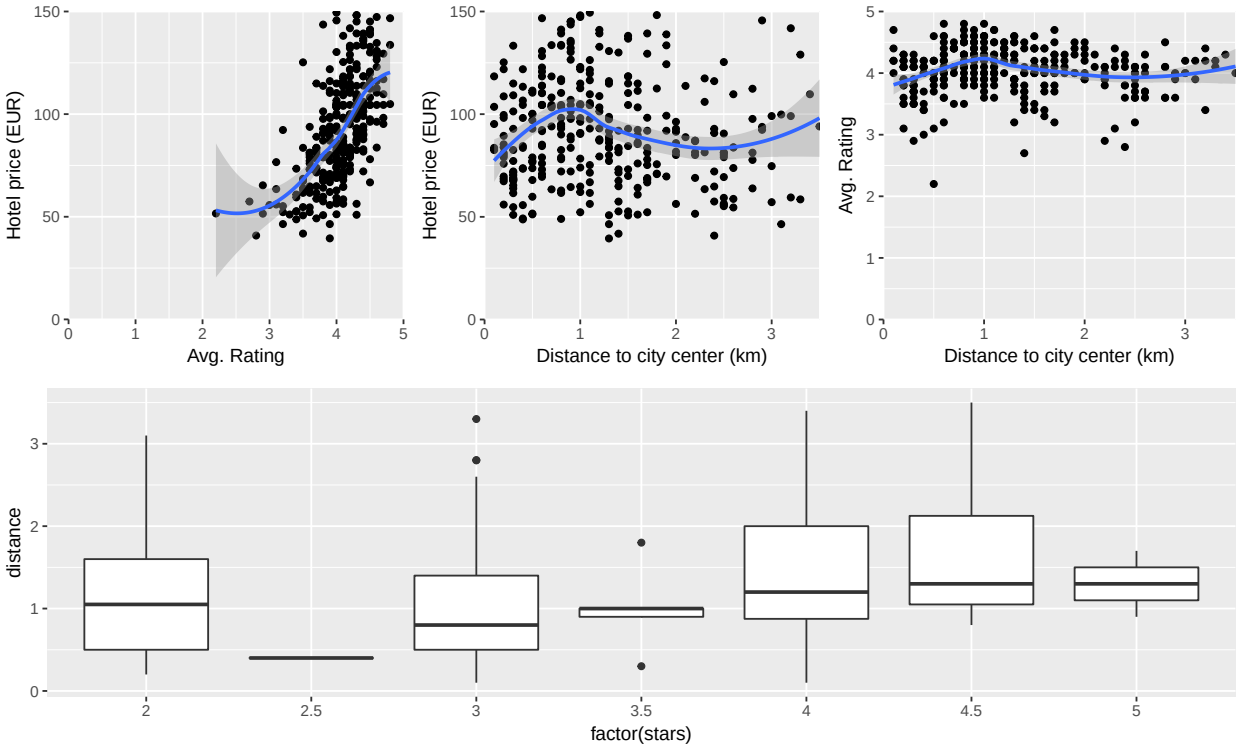
In the next step I'll define a sub-sample of all previous records for further analysis. I admit the choices are made are somewhat arbitrary and they are not aiming to remove errorneous values (something that we'll see in a later question).

The decisions to remove certain records are primaraly due to two reasons:

1. I wanted to build the model in a range where are observations are "dense enough". In certain data ranges, where we don't have enough observations, a simple model can over-react
2. In real-life examples we will always have certain constraints. If we are looking for the best hotel value, we still might have a budget (capped in 150 euros below), and a distance we are willing to walk at the worst case (3.5km below)

```
dt <- dt[price <= 150, ]
dt <- dt[distance <= 3.5, ]
dt <- dt[order(-price), c('name', 'stars', 'rating', 'distance', 'price')]
```

Before we start to build our models, a few more visualizations are useful for additional context:



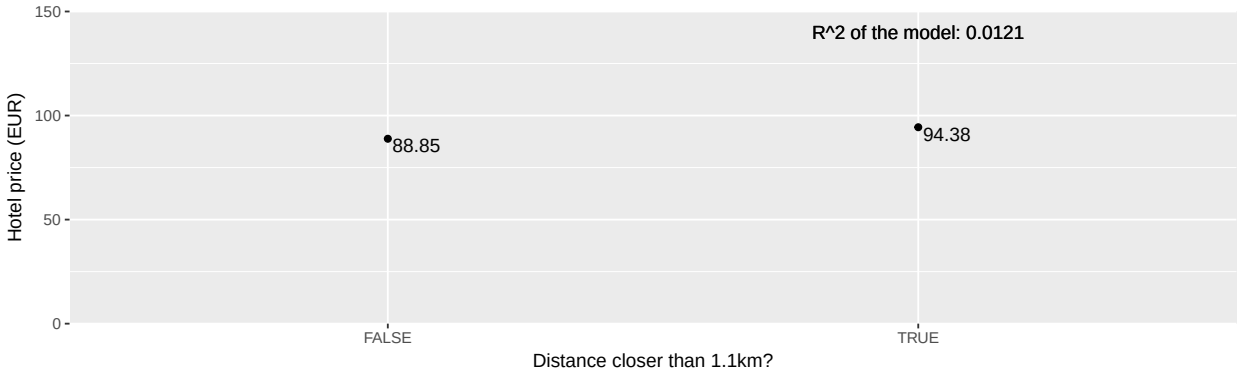
An interesting relationship that we can see is among ratings, prices, and distances. It seems that hotels which are around 1km away from the city center have a bit of a bump in ratings (on average), and a clear positive relationship between ratings and prices. This tells us that distances alone might be insufficient to explain price differences, and more interestingly, that hotels in a certain distance seem to be the best (if we accept ratings and prices as indicators).

3. Create a binary variable of distance (below/above cutoff of your choice) and regress price on this binary variable. Report, interpret and visualize the results. (1-2 sentences)

First, let's create the new variable (I arbitrarly chose the median as cutoff value), and define our simple model:

```
cutoff <- dt[, median(distance)]
dt[, is_close := distance < cutoff]

binary_model <- dt[, .(avg_price = mean(price)), by = is_close]
dt[, binary_pred := mean(price), by = is_close]

r2_binary <- var(dt$binary_pred) / var(dt$price)
```

And now visualize the results and the model's explanatory power:



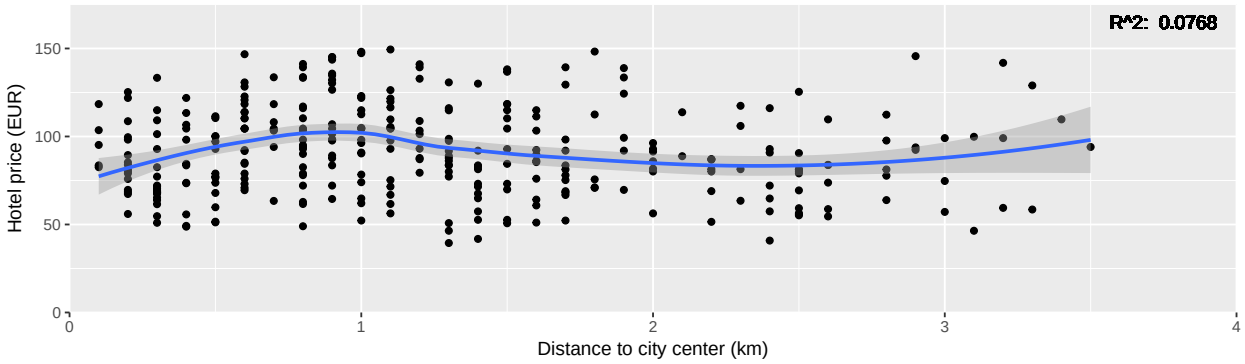
This model is very simple, using two averages to predict price for all distances - not necessarly a bad thing, however we can see that it only explains little to none (1.2%) of the variance in hotel prices. We could improve on this by adding more distance points to predict for (or possibly with a better chosen cutoff value), but it is better worth moving on to more sophistacted models.

4. Estimate a lowess nonparametric regression of price on distance. Report, interpret and visualize the results. (1-2 sentences)

First, create the model and its predictions:

```
loess_model <- loess(price ~ distance, dt)
dt$loess_pred <- predict(loess_model)
r2_loess <- var(dt$loess_pred) / var(dt$price)
```

Visualizations of the results:



If we concentrate on the model's fit, we can see an improvement compared to the previous example, as R^2 increased to 0.0768 - which still indicates a relatively small, but maybe not ignorable correlation between the two variables. There is a drawback to this model however - as it is non-parametric, we can't use it for explanaing the relationships in our data.

The flip side is however that the model is very flexible, and able to capture non-linearities (as seen in the plot above) - I'll be using its R^2 later for ranking other methods.

5. Estimate a simple linear regression of price on distance. Report, interpret and visualize the results. (1-2 sentences)

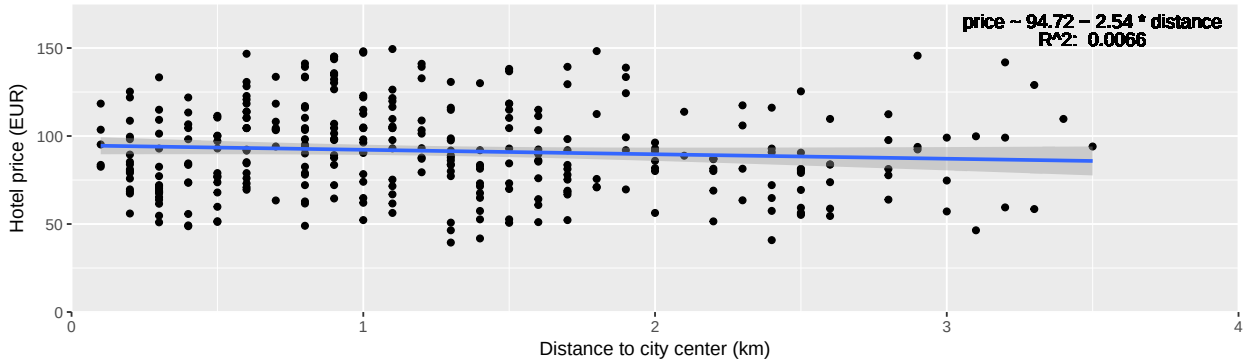
The model & predictions:

```
simpleLM_model <- lm(price ~ distance, dt)
dt$simpleLM_pred <- predict(simpleLM_model)
```

```
r2_simpleLM <- var(dt$simpleLM_pred) / var(dt$price)
simpleLM_formula <- as.formula(
  paste0("price ~ ", round(coefficients(simpleLM_model)[1],2), "",
    paste(sprintf(" %+.2f*%s ",
                  coefficients(simpleLM_model)[-1],
                  names(coefficients(simpleLM_model)[-1])),
          collapse="")
  )
)
format(simpleLM_formula)
```

```
## [1] "price ~ 94.72 - 2.54 * distance"
```

Visualizations of the results:



A good thing about a linear model is that its results are very easily interpretable. Here, we could say every kilometer we go further from the city center is expected to save us 2.5 euros on average in the hotel price. This results is however very weak, the fit is possibly zero, and our line is close to flat.

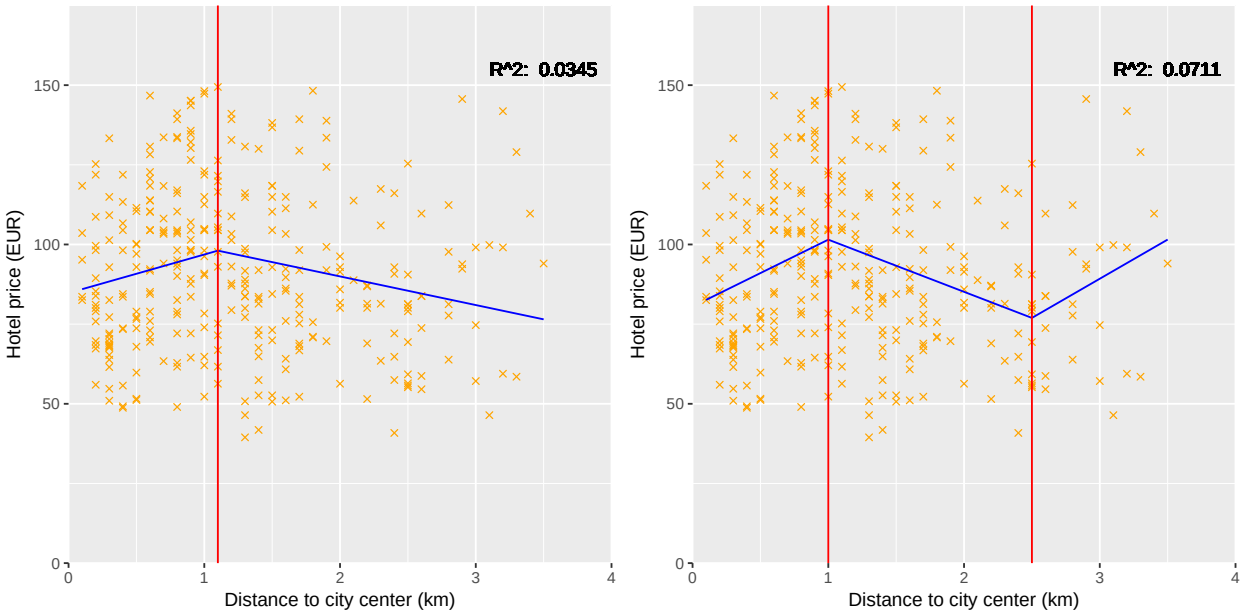
6. Estimate a linear regression of price on distance that captures potential nonlinearities (polynomials, splies). Report, interpret and visualize the results. (1-2 sentences)

First, let's build a spline with 1 and 2 knots.

The models & predictions:

```
##1 knot
spline_1knot_model <- lm(price ~ lspline(distance, cutoff), data=dt)
dt$spline_1knot_pred <- predict(spline_1knot_model)
r2_spline_1knot <- var(dt$spline_1knot_pred) / var(dt$price)


## 2 knots
knots = c(1, 2.5)
spline_2knot_model <- lm(price ~ lspline(distance, knots), data = dt)
dt$spline_2knot_pred <- predict(spline_2knot_model)
r2_spline_2knot <- var(dt$spline_2knot_pred) / var(dt$price)
```

Visualizations of the results:

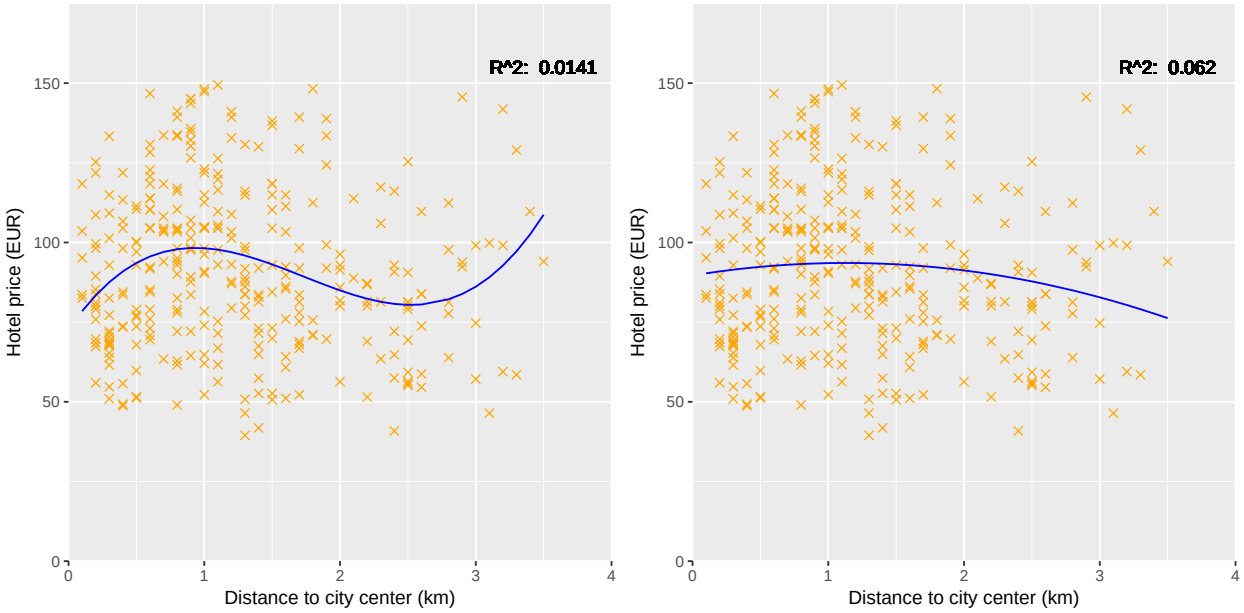



```
##                (Intercept) lspline(distance, cutoff)1
##                  84.762532                  12.090850
## lspline(distance, cutoff)2
##                  -8.984767
```

```
##               (Intercept) lspline(distance, knots)1
##                  80.52459                 20.98863
## lspline(distance, knots)2 lspline(distance, knots)3
##                 -16.37499                 24.59166
```

One of the challanges with using splines is that the knots need to be manually specified - for this example, I tried to use 'turning points' from our earlier lowess model. Analyzing the results we can see that there is indeed a significant change in terms of price ~ distance dynamics in the data around the knots. However, our fit is still weak (both by eye-balling the graphs and seeing the R^2-s). There is some improvement compared to the simple linear regression, however this model would still not be of much use in real life.

We can also try to address non-linearites in the data by adding polinomal terms of our RHS variables. Let's see what happens if we use the square and cubic of distance as well:




```
## (Intercept)    distance distance_sq
##   89.658933    6.936891   -3.076049
```

```
##  (Intercept)     distance  distance_sq distance_cub
##    72.623390    62.723444   -46.011648     8.867607
```

I'm not going to spend much time here as the results are very similar to what we have observed previously: the fit is weak, but we can observe some relationship, that exhibits non-linear traits as well.

7. Discuss your overall findings. (2-3 sentences)

Honestly, I am suprised how little relationship I could uncover between distance and hotel prices in the selected sample for Barcelona.

This could be related to several reasons:

1. unobserved variables (which is not a suprise with only one predictor)
2. the definition of city center - I was plotting the locations on a map, and given Barcelona's dynamic city structure, it could be highly likely that certain parts of the city (e.g. less-good neighborhoods close to the city center) could make the relationship between price and distance very complex
3. the definition of the sample was not optimal for our purpose
4. or simply that there is no explanatory power of distance to city center against hotel prices

What was not suprising is that more flexible models tended to perform better (but still not so well) - however, most of the presented models achieved this with a trade-off in interpretation, something that might or might not be a problem depending on the business case.

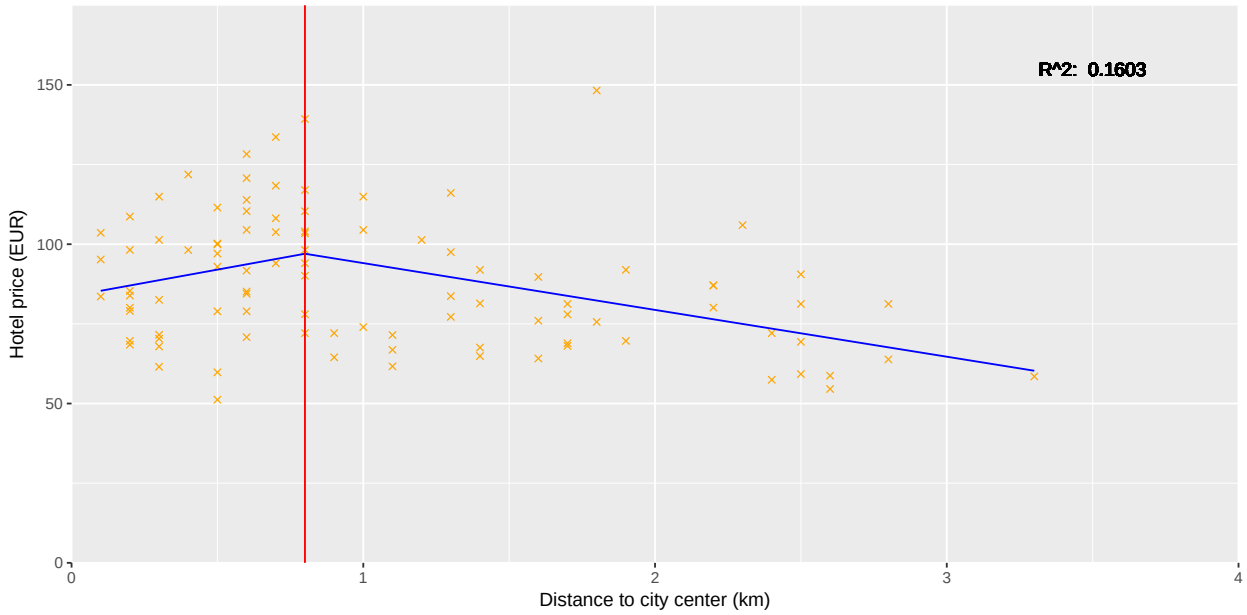
+1: See what happens when you estimate your models on a selected subsample (ie exclude some hotels based on stars, or location). Discuss the role of cleaning and sample selection.

First, let's create the sub-sample based on more specific filtering, building on the plots uncovering the relationship between stars and hotel prices in Question #1. Building a 1 knot spline for this sub-sample:

```
dt <- hotels[accommodation_type == 'Hotel', ]
dt <- dt[rating > 3.5, ]
dt <- dt[city == 'Barcelona', ]
```

```
dt <- dt[distance < 3.5, ]
dt <- dt[stars == 3, ]

spline_subsample_model <- lm(price ~ lspline(distance, 0.8), data = dt)
dt$price_pred <- predict(spline_subsample_model)
r2_spline_subsample <- var(dt$price_pred) / var(dt$price)
```

Visualization of the results:




```
##             (Intercept) lspline(distance, 0.8)1 lspline(distance, 0.8)2
##                83.72509               16.60398              -14.68730
```

I think we can call these results promising. With the same model as we used in a previous excersize, we could increase the fit, R^2 went from 0.03 to 0.16 for the same model type, which we could not disregard as insignificant anymore. Basically, what we achieved is substituting an unobserved variable in our model with a more precisely selected sample.

This example clearly shows the importance of data preparation and sample selection. With a well defined sample, we could achieve what would have been otherwise only possible with a lot more complex model. In real business cases, the data preparation needs to be based on the actual problem being solved. An example is removing far away hotels: they only carry valuable information for us if we are willing to go that far, otherwise they just create unnessary noise in our modeling.

All in all, we can conclude by stating that sample selection and data cleanising is the first step to success in modeling - done right, they will save us from many headaches later on.