Exploratory Data Analysis on Housing Market in Italy

Supervised Learning & Visualization

Daniel Anadria Kyuri Park Ernst-Paul Swens Emilia Löscher

October 3, 2022

Contents

1 Introduction

This is an exploratory data analysis of the Italian housing market in 2022. For context, Italy contains a total of 20 regions (*regioni*), 107 provinces (*province*) and 7,904 municipalities (*comuni*). In the present work, we pose several interesting research questions which can be answered by means of data visualization and predictive model building.

1.1 The Dataset

Our dataset originates from Kaggle. It contains information about the housing market in Italy in 2022. The data were scraped from one of the most prominent housing sales websites in Italy during the month of *August 2022*. The data consist of more than 223,000 sales posts spread over 7,023 (89% coverage) Italian municipalities. We do not have any information on the representativeness of our dataset. Hence, we advise caution when drawing inferences from our findings.

In order to plot the statistics of interest to maps of Italy, we use the regional and provincial shape files, which are obtained from the *Italian National Institute of Statistics* (ISTAT). These files contain the regional and provincial coding and geographical shape information, which can be used to cluster the municipalities in our `location` variable into their respective provinces and regions.

For each housing sale post, the dataset contains the following variables:

Table 1: Description of Variables in the Italy Housing Dataset

Variable	**Description**
`id`	ID of the sale
`timestamp`	Timestamp consisting of 10 digits
`location`	Location on municipality level
`title`	Short description of property
`price`	Price in Euros
`n_rooms`	Number of rooms
`floor`	Floor
`mq`	Size in square meters
`n_bathrooms`	Number of bathrooms
`year_of_construction`	Year of construction
`availability`	Availability of property
`energy_class`	Energy class ranging from a+ to g
`status`	Status of the property
`heating`	Type of heating
`has_garage`	Garage present: yes (1), no (0)
`has_terrace`	Terrace present: yes (1), no (0)
`has_garden`	Garden present: yes (1), no (0)
`has_balcony`	Balcony present: yes (1), no (0)
`has_fireplace`	Fireplace present: yes (1), no (0)
`has_alarm`	Alarm present: yes (1), no (0)
`has_air_conditioning`	Air Conditioning present: yes (1), no (0)
`has_pool`	Pool present: yes (1), no (0)
`has_parking`	Parking present: yes (1), no (0)
`has_elevator`	Elevator present: yes (1), no (0)
`is_furnished`	Furniture present: yes (1), no (0)

2 Preparation

In order to start our exploratory analysis, we first load relevant packages and import the dataset as well as the ISTAT shape files.

2.1 Load Packages & Import Data

```r
## load packages
library(tidyverse)   # for wrangling data
library(magrittr)    # for using pipes
library(skimr)       # for skimming data
library(sf)          # for spatial analysis
library(sp)          # for spatial analysis
library(ggplot2)     # for plotting
library(fuzzyjoin)   # for joining on not-exact matches
library(ggpubr)      # for arranging ggplots
library(mice)        # for imputation procedure

## import italy housing data
houses <- read.csv("data/housing_data_italy_august2022.csv",
                    na.strings=c("","NA"), header = TRUE)

## import istat shape files
# municipality
municipalities <- st_read("data/italy_shape_2022_files/Com01012022_g")
municipalities <- municipalities[c("COD_REG", "COD_PROV", "COMUNE")]
# province
provinces <- st_read("data/italy_shape_2022_files/ProvCM01012022_g")
# region
regions <- st_read("data/italy_shape_2022_files/Reg01012022_g")
```

3 Exploratory Research Analyses

We focus on following the exploratory analyses:
1. Exploring the missingness in the dataset. 2. Investigate whether there any geographical trends in the median housing prices and their absolute deviations on regional and/or provincial level?

4 Data Cleaning

Note: We base the following data cleaning on the summary of the raw dataset, which can be found in the *Appendix*.

The original data consist of 223,409 rows (*sales*) and 25 columns (*variables*).
Given our research questions, we exclude `id` (*ID of the sale*), `timestamp` (*timestamp of the sale*), and `title` (*description of the property*) as they are deemed irrelevant. In addition, we exclude two columns that have only one unique value (`status` and `availibility`), as these are not variables but constants.

We observe that types of some variables are wrongly specified. We convert them to a correct type (e.g., `heating`: character→factor, `has_xxx`: numeric→factor, `is_furnished`: numeric→factor).

Next we create a new variable `property_age` by subtracting the `year_of_construction` from 2022. In the original dataset, there are some unreasonable years of construction (e.g., 2209). While some properties may be sold before their construction is completed, we deem it unlikely for properties whose `year_of_construction` is more than 4 years later as of now. Thus, we filter out those with `year_of_construction` > 2026.

The variable of our main interest `price` is highly skewed to the right given that the mean (239,939) is far off to the right of the median (135,000). We take a closer look at the distribution of the `price` with the help of a boxplot to examine the outliers (see *Figure 1*).

```
## create our own theme that can be used throughout
custom.theme = theme(
  axis.title.x = element_text(size = 14),
  axis.text.x = element_text(size = 13),
  axis.title.y = element_text(size = 14),
  axis.text.y = element_text(size = 13))

## boxplot of price
houses %>%
   ggplot(aes(x=price)) +
   geom_boxplot() +
   # add comma on the x-axis labels
   scale_x_continuous(labels=scales::label_comma(),
   # rotate the x-axis labels
   guide = guide_axis(angle = 25)) +
   # apply minimal theme plus our own theme
   theme_minimal() + custom.theme
```

From *Figure 1*, we observe that there are extreme outliers in `price`. Some housing prices in the dataset are exorbitant (e.g., over €2B). We decide to focus the scope of our analysis on the houses whose price is less than or equal to €1M, which are more likely to be affordable to an average Italian. The distribution of housing prices after filtering can be seen in *Figure 2*.

```
## density plot of price (cleaned dataset)
houses %>%
  # filter the price over a million
  filter(price <= 1e6 | is.na(price)) %>%
  # create a ggplot
  ggplot(aes(price)) +
  # add histogram
  geom_histogram(aes(y=..density..), bins = 30, color = 1, fill="white") +
  # add density line
  geom_density(lwd=0.5, color = "#165e70", fill = "#165e70", alpha = 0.2) +
  # apply our theme
  theme_minimal() + custom.theme
```

From *Figure 2*, we take that the distribution of housing prices after filtering appears a lot more ordinary. There is still a long right tail, but that is to be expected with housing prices in any country. The extreme

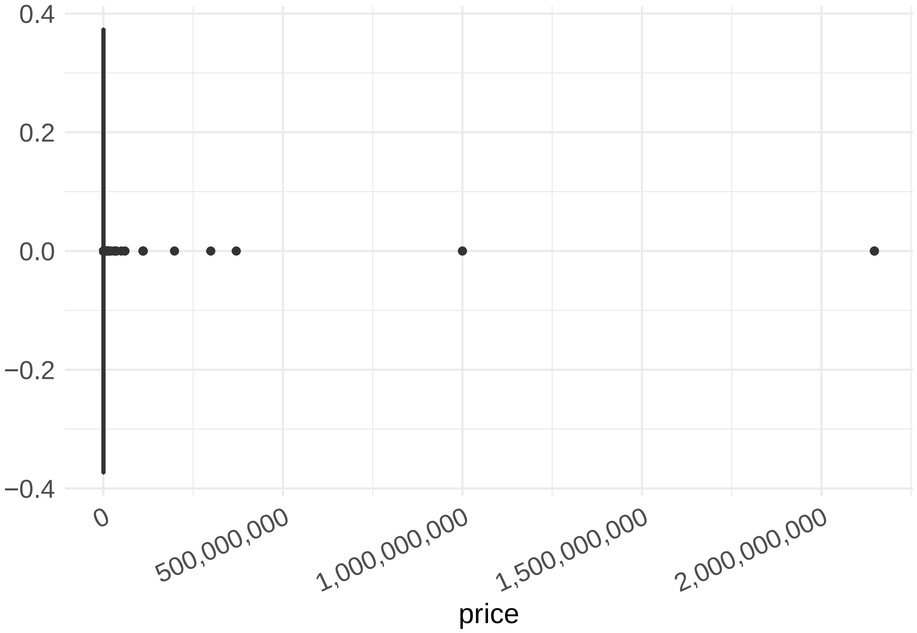


Figure 1: Boxplot of Housing Prices in Italy

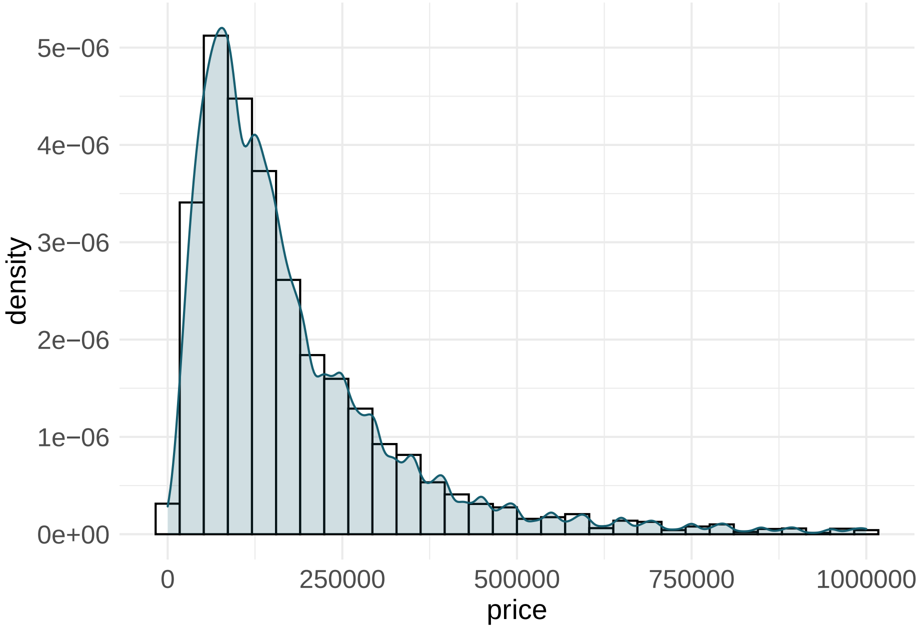


Figure 2: Histogram and Density Plot of Housing Price After Filtering

outliers have been eliminated. From this plot, we also conclude that when working with housing price data, it is likely to be more informative to use centrality and spread measures that are robust to skewed data. For this reason, we will use the *median* and *median absolute deviation (MAD)* instead of the mean and variance in our exploration of the present dataset.

```r
## cleaning up the housing data
houses.cleaned <- houses %>%
  # only select variables that have more than one unique value
  select(where(~n_distinct(.) > 1)) %>%
  # fix the data type (convert them to factor)
  mutate(across(c(starts_with("has"), is_furnished, heating, energy_class, n_rooms,
                  n_bathrooms, location), factor)) %>%
  # filter out houses whose price is over a million (while keeping NAs)
  filter(price <= 1e6 | is.na(price),
  # filter out houses with a built year over 2026 (while keeping NAs)
         year_of_construction < 2026 | is.na(year_of_construction)) %>%
  # create property age variable
  mutate(property_age = 2022 - as.numeric(year_of_construction)) %>%
  # remove id, timestamp, title and year_of_construction
  select(-c(id, timestamp, title, year_of_construction))
```

4.1 Data Summary

After data cleaning, we take a look at the summary statistics to get a better overview of our data. We skim through our cleaned dataset using the `skimr` package.

```r
# settings: round up by 2 decimal places & disable scientific notation
options(digits = 3, scipen = 999)
# specify skimming function
custom.skim <- skim_with(character = sfl(whitespace=NULL, min = NULL, max = NULL),
   factor = sfl(ordered = NULL),
   numeric = sfl(p0 = NULL, p25=NULL, p50=NULL, p75=NULL, p100=NULL, hist=NULL,
                 median = ~median(., na.rm=T),
                 min = ~min(., na.rm=T), max = ~max(., na.rm=T), n_unique=n_unique))
# summary table
custom.skim(houses.cleaned)
```

Table 2: Data summary

Name	houses.cleaned
Number of rows	220748
Number of columns	20
Column type frequency:	
factor	16
numeric	4
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	n_unique	top_counts
location	0	1.00	7023	pis: 192, leg: 190, la : 188, bar: 187
n_rooms	58297	0.74	4	3: 56766, 4: 47144, 5: 30944, 2: 27597
n_bathrooms	12677	0.94	3	1: 107372, 2: 79843, 3: 20856
energy_class	638	1.00	12	g: 115238, f: 25396, e: 17124, a: 15931
heating	0	1.00	2	aut: 197849, oth: 22899
has_garage	0	1.00	2	0: 180669, 1: 40079
has_terrace	0	1.00	2	0: 196111, 1: 24637
has_garden	0	1.00	2	0: 184426, 1: 36322
has_balcony	0	1.00	2	0: 198058, 1: 22690
has_fireplace	0	1.00	2	0: 208817, 1: 11931
has_alarm	0	1.00	2	0: 218752, 1: 1996
has_air_conditioning	0	1.00	2	0: 155058, 1: 65690
has_pool	0	1.00	2	0: 216473, 1: 4275
has_parking	0	1.00	2	0: 217364, 1: 3384
has_elevator	0	1.00	2	0: 208067, 1: 12681
is_furnished	0	1.00	2	0: 203644, 1: 17104

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	median	min	max	n_unique
price	39113	0.82	177784.71	153812.51	130000	1	1000000	2555
floor	71398	0.68	1.82	1.13	2	1	52	20
mq	3343	0.98	156.08	124.68	116	1	999	971
property_age	10	1.00	56.07	74.36	42	-3	1022	375

From the output, we see that our cleaned dataset has 220,607 rows and 20 columns, 16 of which are factors, and 4 of which are numeric types. The output is presented in two tables for factor and numeric variables, separately.

From the table of factor variables, we again see that `location` has 7023 unique values (i.e., municipalities). Also, there are some missing values for `energy_class` and a lot of missing values for `n_rooms` and `n_bathrooms`.

From the table of numeric variables, we see that all numeric variables have some missing values. For about 18% of `price`, the variable of our main interest, is missing and we look into this more in detail when we address the 2nd question regarding the missingness in `price`. ***@ALL: Any other statistics interesting? mean, sd?.. something?***

5 Exploratory Data Analysis

In the following we are performing the exploratory data analyses. First, we explore the missingness in the dataset. Afterwards, we explore trends in *house price* on the regional and provincial level.

5.1 Question 1: Exploring missingness in the dataset.

As we later want to analyze the data on the regional and provincial level, we also want to explore the missingness on these levels. Therefore, we aggregate the data from a municipality level to the regional and provincial level. This process is described in the following section.

5.2 Geographical Data Preparation

At the beginning of the assignment, we loaded the *ISTAT* shape files. These files are useful for two reasons. First, they contain the list of all Italian municipalities, their respective provinces and regions. Therefore, we can use this data to append our original dataset with additional location indicators. Second, they contain the shapes of Italy divided into provinces and regions. This is particularly useful for creating map plots using `ggplot2`.

Each sale in our dataset is assigned to one of 7023 municipalities. In order to create plots which visualize the differences in average housing prices across Italy, we assign each municipality to its corresponding province and region. We use the data from *ISTAT* to append the province and region information to every observed municipality in our dataset. We use fuzzy matching for inexact matches as we found that there were some minor inconsistencies in how the municipalities were named in our dataset as opposed to their names in the ISTAT shape files. The result of the following chunk of code is that all the municipalities are assigned their regions and provinces.

```
## load imputation results
imputations <- readRDS("data/imputation.rds")
```

```
# load the imputed data including the geospatial data
imputed <- readRDS("data/imputed.rds")

# load the cleaned house data including the geospatial data
houses.cleaned <- readRDS("data/housescleaned.rds")
```

To answer our first research question, we aggregate our data on two levels: 1) regional and 2) provincial level by computing two aggregated statistics: 1) the median housing price and 2) the median absolute deviation (MAD) in housing price on the two respective levels. This yields two sub-datasets, one per each level. To each, we attach geometric information needed for geographic plotting and convert it to an `sf` object which is a requirement for plotting maps.

Having done this, we are ready to explore the missingness in the dataset.

First, we do so by calculating and plotting the proportion of missing values for all the variables in Figure @ref(fig:missingness).

```
# barplot to show the missingness per variable
houses.cleaned %>%
   # get the average missing proportion
   is.na() %>% colMeans() %>% stack() %>%
   # create ggplot for the missing proportion
   ggplot(aes(x = reorder(ind, values), y=values)) +
   # specify the bar plot
   geom_bar(stat="identity", width = .2, fill = "black",
            position = position_dodge2(padding = 0.8)) +
   # flip the bar plot and add the points
   geom_point() + coord_flip() +
   # apply our custom theme
   theme_minimal() + custom.theme +
   # specify axis labels
   labs(x = "", y = "Missingness (%)")
```

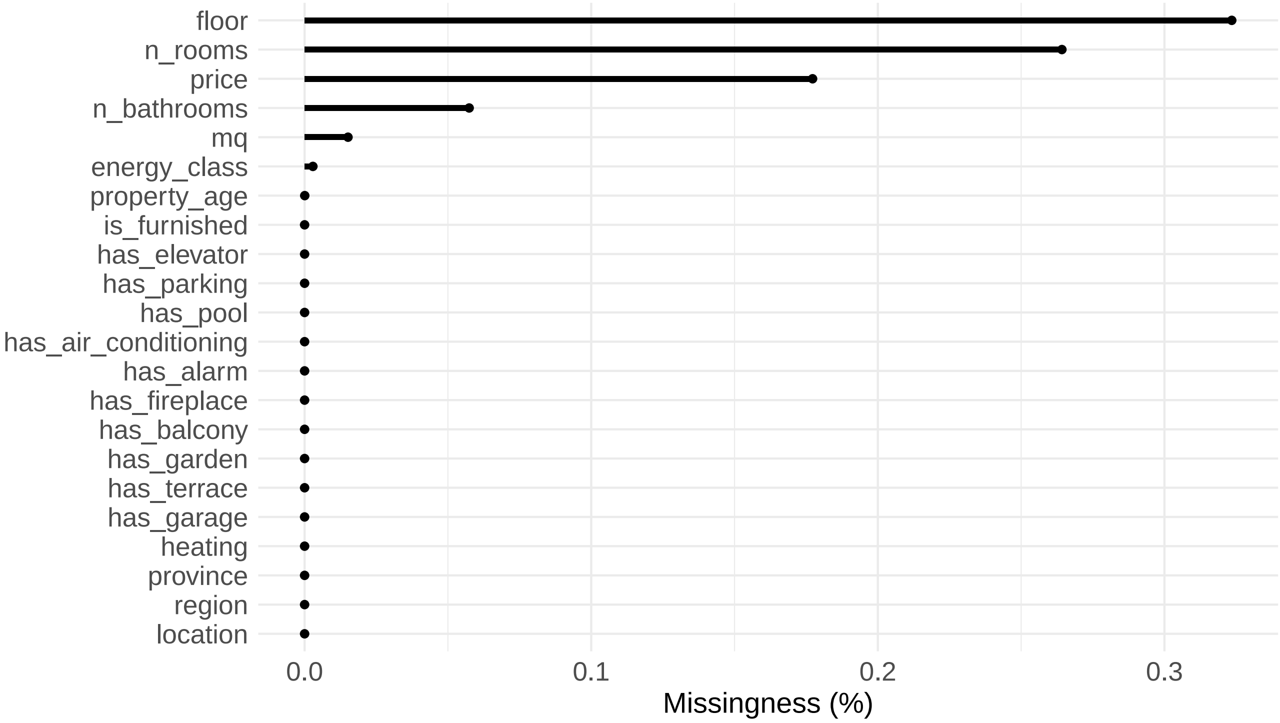


Figure 3: Proportion of Missing Values

The plot shows that the percentage of missing values lies above 30% for the variable *floor* and above 25% for the *number of rooms*. Furthermore, about 18% of the *housing prices* are missing. Except for the *number of bathrooms* which has about 6% values missing, for the remaining variables less than 3% are missing, respectively.

In the following, we explore the correlation of *price* and the other variables and summarize the findings in *Table XX*.

```
houses.cleaned %>%
  # create missingness indicator of price
  mutate(na_ind = ifelse(is.na(price), 1, 0)) %>%
  ungroup() %>%
  # compute correlation
  summarize(across(n_rooms:property_age, ~cor(as.numeric(.x), na_ind,
                                               use = "pairwise.complete.obs"))) %>%
  # sort them in descending order based on their absolute values
  t(.) %>% .[order(abs(.), decreasing = TRUE),] %>%
  # create a neat table
  knitr::kable(col.names = "correlation", caption = "Correlation between missingness
              of price and other variables", align="c")
```

Table 5: Correlation between missingness of price and other variables

	correlation
energy_class	-0.271
has_garden	-0.114
heating	0.109
has_garage	-0.097
has_fireplace	-0.087
has_air_conditioning	-0.085
has_terrace	-0.078
has_elevator	-0.077
n_rooms	0.062
has_balcony	-0.056
has_parking	-0.038
mq	0.034
has_alarm	-0.032
property_age	-0.020
n_bathrooms	0.017
floor	0.010
has_pool	-0.010
is_furnished	0.005

The missingness of `price` appears to be moderately correlated with the `energy_class` (*cor = -0.271*). Hence, we further check what the pattern of missingness in price across different energy classes looks like.

```r
## create plots for missingness in price vs energy class
houses.cleaned %>%
  # add price missingness indicator
  mutate(na_ind = ifelse(is.na(price), 1, 0)) %>%
  # group by energy class
  group_by(energy_class) %>%
  # sum up all the missingness in price per energy class
  summarize(n = sum(na_ind)) %>%
  # create a ggplot for the sum of missingness
  ggplot(aes(x = energy_class, y = n, fill = energy_class)) +
  # turn off legend
  geom_col(show.legend=FALSE) +
  # customize the color
  scale_fill_manual(values = c("#F8766D", "#999999","#F8766D",
                                  rep("#999999", 8),"#F8766D")) +
  # apply the themes to the bar plot
  theme_minimal() + custom.theme +
  # change the labels
  labs(x = "Energy Class", y = "Missingness in price")
```

Figure XX shows that there is higher missingness in price for houses that either have a good energy class

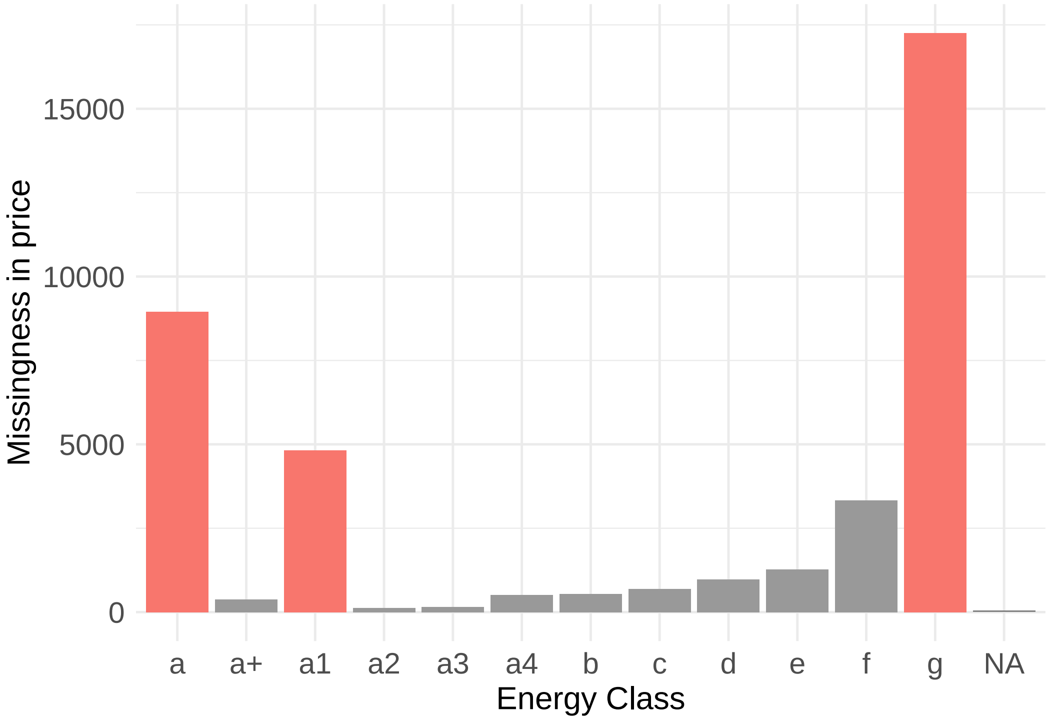


Figure 4: Missing Values in Price across Different Energy Classes

of *a* or *a1*, or fall into the very inefficient energy class *g*. There is low missingness in *price* for the other energy classes and for houses for which *price* is missing.

We also check the missingness in price across different regions to see if there are any patterns.

```
## check missingness in price w.r.t regions
houses.cleaned %>%
   # add price missingness indicator
   mutate(na_ind = ifelse(is.na(price), 1, 0)) %>%
   group_by(region) %>%
   # get the total missingness proportion per region
   summarize(`Average missing proportion (%)` = sum(na_ind) / n()) %>%
   # add the spatial data
   left_join(., regions, by = c("region" = "DEN_REG")) %>%
   st_as_sf() %>%
   ggplot() +
   # plot the italy map
   geom_sf(fill=NA) +
  # add scatter plots of missingness proportion per region
   geom_point(color = alpha("red", 0.4),
   aes(size = `Average missing proportion (%)`, geometry = geometry),
   stat = "sf_coordinates") +
  # remove unnecessary coordinates
   theme_void() +
   theme(legend.position = "bottom")
```

Figure 7 shows that the proportion of missing data in `price` is not equal across different regions.

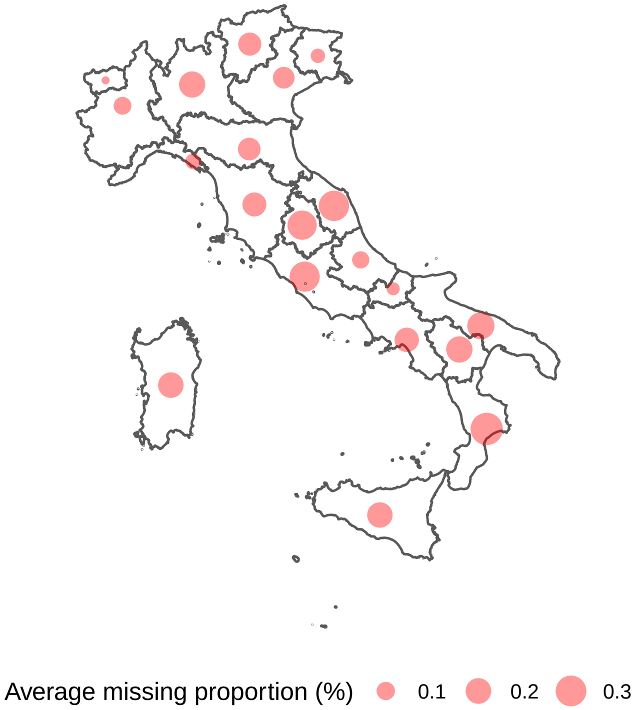


Figure 5: Missingness of Price per Region

Next, we examine the relationship between the predictors' missingness indicators and the *price* densities by plotting the density of price (on a log10-scale) split by whether or not the respective value for a given variable is missing or not. These results provide us with some indication of the missing data mechanism. Specifically, if missing completely at random (MCAR) is plausible to assume or not.

```r
## function to create a density plot of price with missing indicator
missing.plots <- function(x) {
   plot <- houses.cleaned %>% ungroup() %>%
      mutate(missing = is.na(.[,x])) %>%
      ggplot(aes(x = log10(price), fill = missing)) +
      geom_density(alpha = 0.5, color = NA) +
      theme_classic() +
      labs(x = expression(paste(Log["10"],"(Price)"), y = "Density")) +
      scale_x_continuous(limits = c(2, 7)) + ylim(0, 1.5)
   return(plot)
}

# multiple plots for floor, no. of rooms, no. of bathrooms, and meters squared
ggarrange(missing.plots("floor"),
          missing.plots("n_rooms"),
          missing.plots("n_bathrooms"),
          missing.plots("mq"),
          labels = c("         A. Floor", "B. Number of rooms", "C. Number of bathrooms", "D. Squ
                     font.label=list(color="black",size = 10),
          ncol = 2, nrow = 2, common.legend = TRUE, legend = "bottom")
```

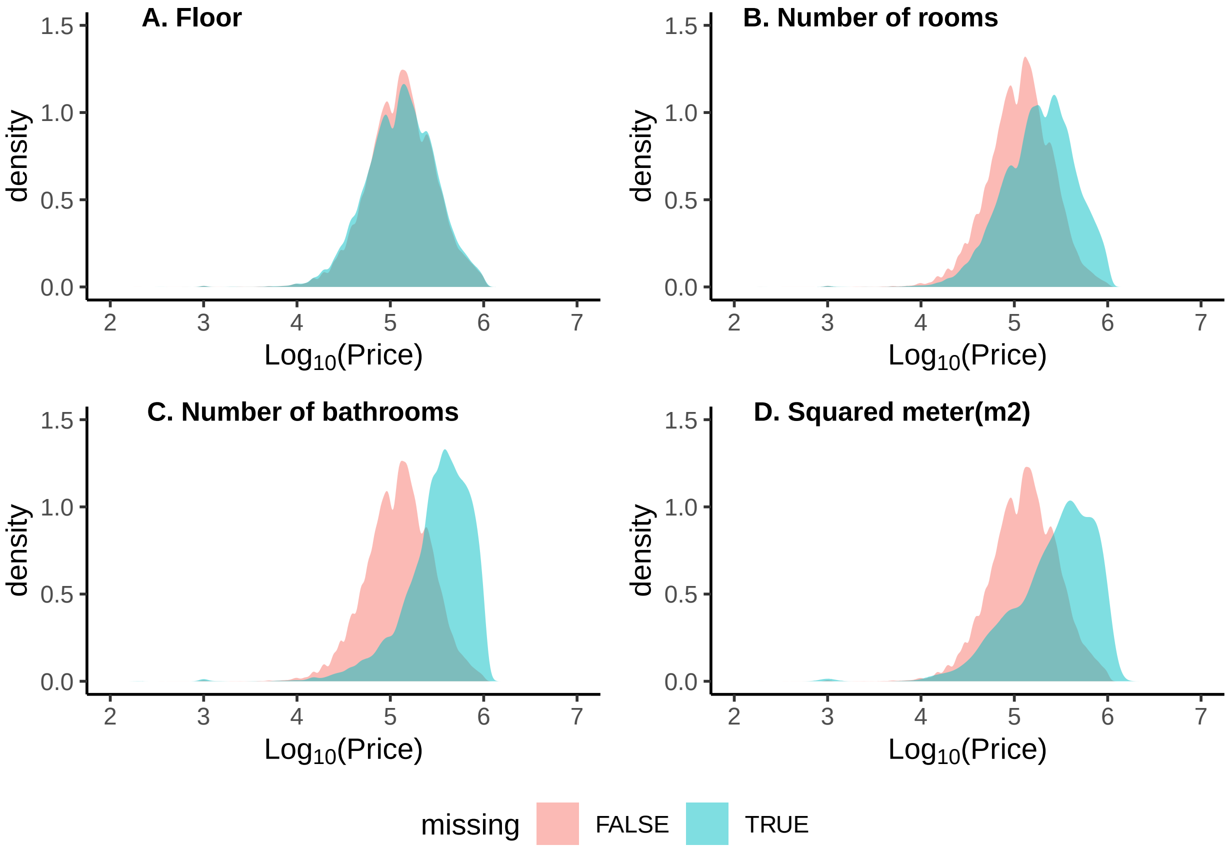


Figure 6: Density Plots for Price based on the Missing Indicators

The missingness of number of floors (@ref(fig:missingness_density) A) and number of rooms (@ref(fig:missingness_density) B) does not seem to be dependent on the observed price information. Whereas for the number of bathrooms (@ref(fig:missingness_density) C) and the meters squared (@ref(fig:missingness_density) D) missingness shows a different result, namely, missingness tends to occur at higher house prices.

5.2.1 Intermediate Conclusion

We explored if the missingness of the predictors and the outcome variable *house prices* are related. From our results it appears that they are. Hence, it is unlikely that the missingness mechanism is MCAR. Therefore, we include a multiple imputation procedure before investigating geographical trends in the median and the MAD.

5.2.2 Imputation

Furthermore, we limited the diagnostics for the imputation procedure to the convergence of the algorithm and plausibility of the imputed data. There appeared to be no convergence issues and the imputed data appeared to be plausible with respect to the observed data (see Appendix).

1st Question : SECTION CONCLUSION / DESCRIPTION

5.3 Question 2: Regional and Provincial Trends in the Median Housing Price and the Median Absolute Deviations in Italy

We use the imputed dataset to explore whether there is a geographical trend in the median and the median absolute deviation (MAD) *housing price* on a regional or provincial level. In order to do so, we plot the median and the MAD on the regional and the provincial level:

```r
## function to get aggregated region price information
group.region <- function(data, estimate) {
   data %>%
      group_by(region) %>%
      summarize(estimate = ifelse(estimate, median(price), mad(price))) %>%
      select(-region)
}

region.median <- lapply(imputed, group.region, estimate = T) %>%
   do.call(cbind, .) %>% rowMeans()
region.mad <- lapply(imputed, group.region, estimate = F) %>%
   do.call(cbind, .) %>% rowMeans()

price.by.region <- data.frame(region = sort(regions$DEN_REG),
   median = region.median, mad = region.mad) %>%
   left_join(.,regions, by = c("region" = "DEN_REG")) %>% st_as_sf()
```

```r
## function to get aggregated province price information
group.province <- function(data, estimate) {
   data %>%
      group_by(province) %>%
      summarize(estimate = ifelse(estimate, median(price), mad(price))) %>%
      select(-province)
}

province.median <- lapply(imputed, group.province, estimate = T) %>%
   do.call(cbind, .) %>% rowMeans()
province.mad <- lapply(imputed, group.province, estimate = F) %>%
   do.call(cbind, .) %>% rowMeans()

price.by.province <- data.frame(province = sort(provinces$DEN_UTS),
   median = province.median, mad = province.mad) %>%
   left_join(., provinces, by = c("province" = "DEN_PROV")) %>% st_as_sf()
```

```
plot.list.1 <- list()

## median & mad of price per region
plot.list.1 <- map(
  c("median", "mad"),
  function(var) {
    ggplot(price.by.region) +
      # map each statistic
      geom_sf(aes(fill = .data[[var]])) +
      # void theme: remove all unncessary coordinates
      theme_void() +
      # color-scheme (color-blind friendly???)
      scale_fill_viridis_c(option = "E", direction = -1) +
      # lengthen the legend
      theme(legend.key.width= unit(2, 'cm'))
     }
  )
plot.list.2 <- list()

## median & mad of price per province
plot.list.2 <- map(
  c("median", "mad"),
  function(var) {
    ggplot(price.by.province) +
      # map each statistic
      geom_sf(aes(fill = .data[[var]])) +
      # void theme: remove all unncessary coordinates
      theme_void() +
      # color-scheme (color-blind friendly???)
      scale_fill_viridis_c(option = "E", direction = -1) +
      # lengthen the legend
      theme(legend.key.width= unit(2, 'cm'))
     }
  )

# combine the plot lists
```

```
plot.list <- c(plot.list.1, plot.list.2)
```

```
## plot the median of price
ggarrange(plotlist = plot.list[c(1,3)], nrow = 1, ncol = 2, common.legend = TRUE,
          legend = "bottom")
```

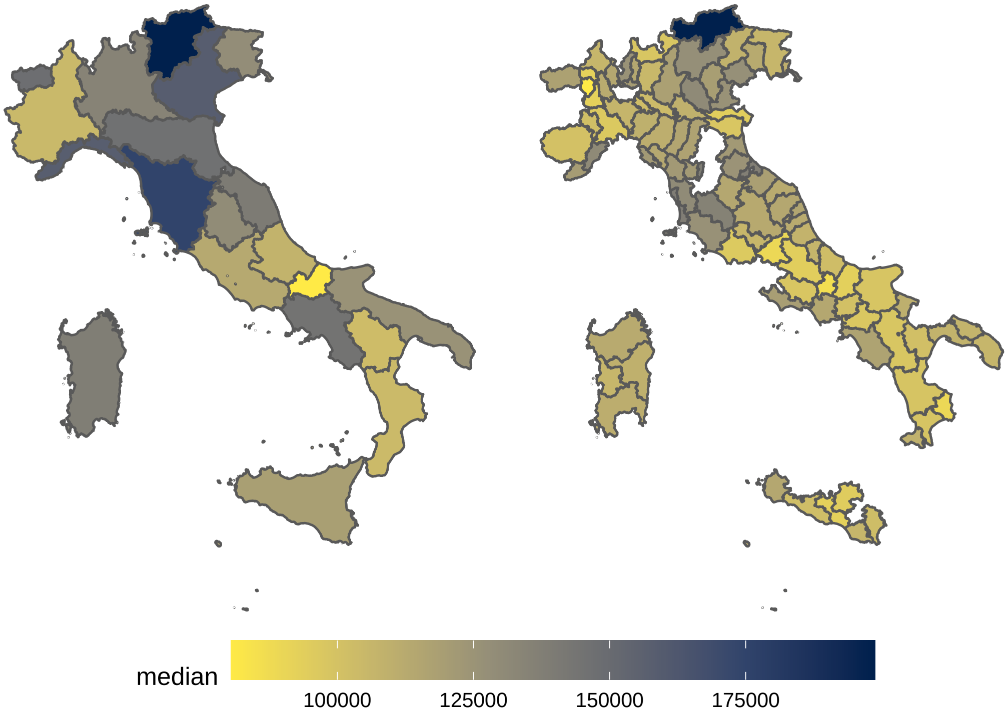


Figure 7: Median of Price per Region (left) and Province (right)

```
## plot the mad of price
ggarrange(plotlist = plot.list[c(2,4)], nrow = 1, ncol = 2, common.legend = TRUE,
          legend = "bottom")
```

On the regional level, we see that the median for the region *Trentino-Alto Adige* (200 000€) are the highest, and the lowest for *Molise* (79 000€). We recognize a trend that the median price is lower for more Southern regions in Italy. The only exception from this is the region of *Piemonte* (99 000€) which has a lower median price than the surrounding regions in the North. Regarding the MAD, a measure of variability within a region, we see that it is highest for the regions with higher median prices. This is recognizable as the color patterns in the median plot and the MAD plot are very similar.

On the provincial level, it can be seen that the high median of the *Trentino-Alto Adige* region is mainly due to the high median of 400 000€ in the province of *Bolzano*. As the other provinces in that region have lower median prices, the MAD in that region is comparably high. The opposite is the case within the Southern regions, here, the provinces that make up the region of *Calabria* for example, all have a low median price. Hence, the MAD for that region is low.

Given that the overall geographical pattern for median and MAD of `price` correspond to each other, it is interesting to investigate further the possible differences in the distribution of `price` between high- and low-median regions.

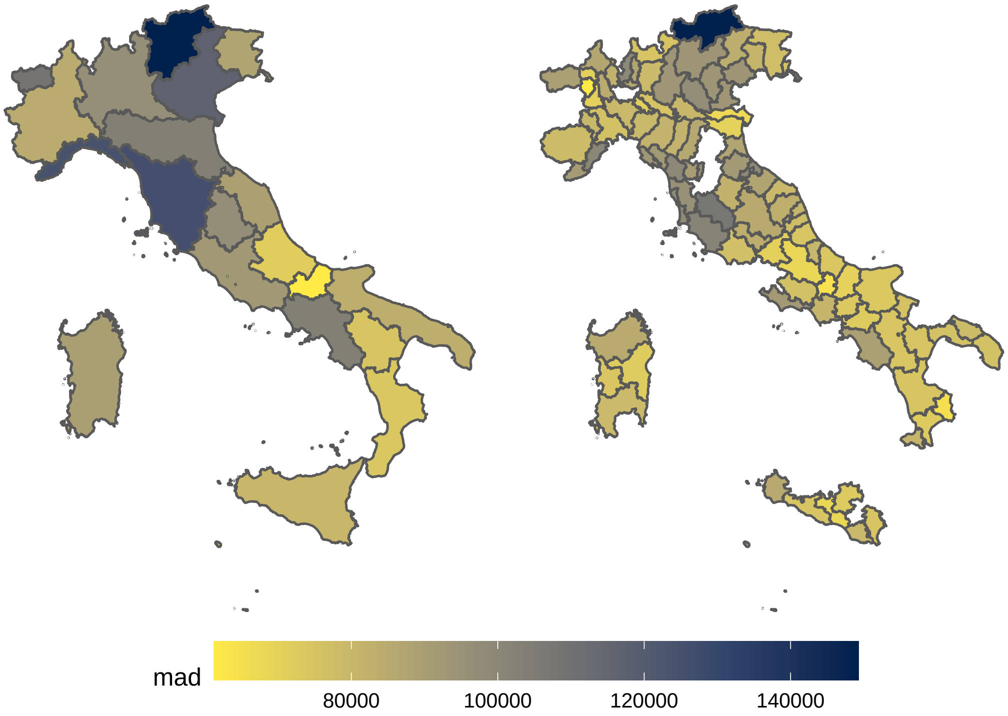


Figure 8: MAD of Price per Region (left) and Province (right)

```
# top 2 high median regions
top.2.median <- price.by.region %>% slice_max(median, n = 2) %>% pull(region)

# lowest 2 low median regions
lowest.2.median <- price.by.region %>% slice_min(median, n = 2) %>% pull(region)

# plot the histograms for each region in the high and low median groups
imputed[[1]] %>%
  # subset top two and bottom two countries
  filter(region %in% c(top.2.median, lowest.2.median)) %>%
  # group by the regions
  group_by(region) %>%
  # create the grouping variable for coloring
  mutate(grouping = ifelse(region %in% top.2.median, "high median", "low median"),
         # get the median price for each region
         med_price = median(price, na.rm=T)) %>%
  # create ggplot for price (coloring by groups)
  ggplot(aes(x = price, fill = grouping)) +
  geom_histogram(bins = 30) +
  # create a panel of plots per region
  theme_bw() + facet_wrap(~region) +
  # indicate the median price by a vertical line
  geom_vline(aes(xintercept = med_price, group=region), linetype="dashed") +
  # change legend title
  labs(fill = "high/low regions")
```

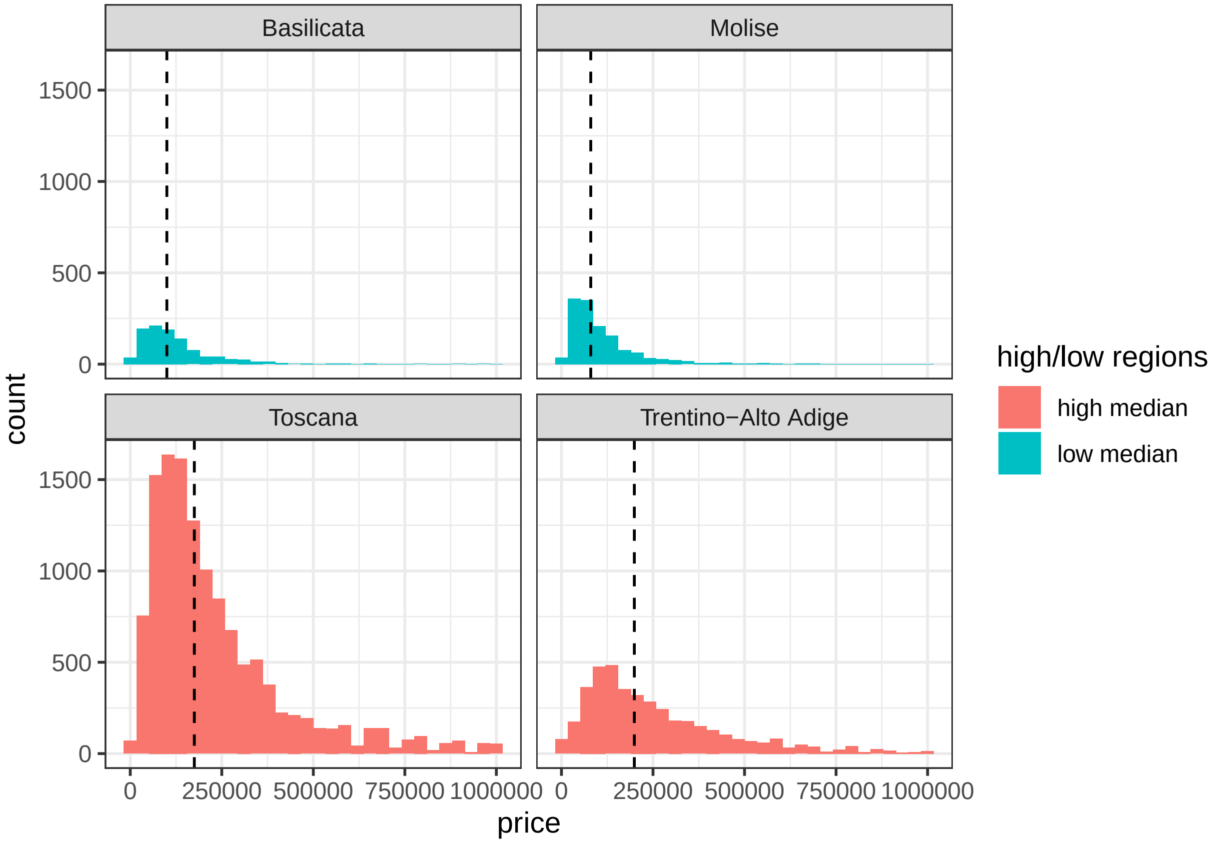


Figure 9: group differences in price distribution

1nd Question : SECTION CONCLUSION / DESCRIPTION An interesting take-away from this figure is that all the densities for the *house price* are right skewed. This is reasonable for house prices as one would expect that there are more cheap and moderately priced houses and only few very expensive houses. Furthermore, it is apparent that there are most sells in the dataset from the *Toscana* region and there are only very few very expensive houses in the regions of *Calabria* and *Molise* as one would expect with those regions having a lower median.

5.3.1 Conclusion

6 Overall Conclusion

All things considered, we identified that there is quite some missingness present - especially for *house price* for which about 18% of values are missing. We were successful in imputing those and the values of the other variables. Using the imputed dataset, we found a trend in the median price from more expensive to cheaper when going from Northern to Southern in Italy. We also have shown that these higher medians on a regional level are due to a higher median on the province level and that in such regions where there is one province with a higher median of *house price*, the MAD is consequently higher too if the other provinces in that region do not have a high median, too.

7 Appendix

7.1 Summary of the raw data using the my_skim function.

```
custom.skim(houses)
```

Table 6: Data summary

Name	houses
Number of rows	223409
Number of columns	25
Column type frequency:	
character	6
numeric	19
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	empty	n_unique
location	0	1	0	7023
title	0	1	0	199305
availability	0	1	0	1
energy_class	679	1	0	12
status	0	1	0	1

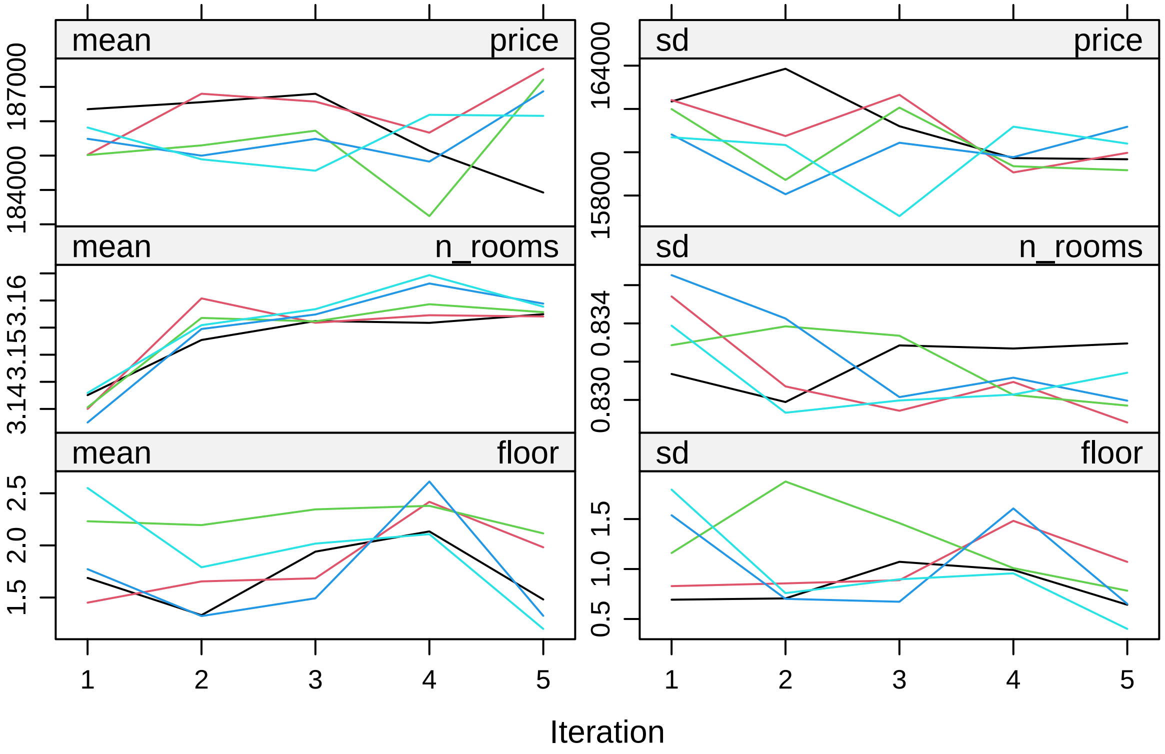
skim_variable	n_missing	complete_rate	empty	n_unique
heating	0	1	0	2

Variable type: numeric

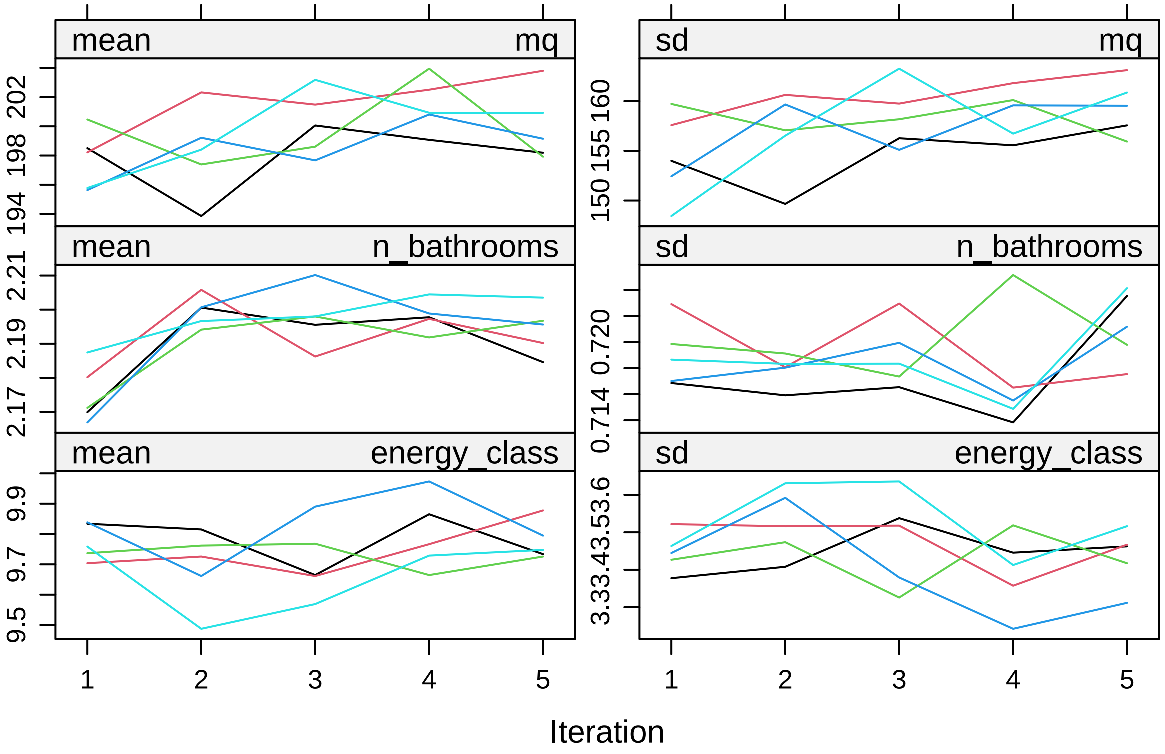
skim_variable	n_missing	complete_rate	mean	sd	median	min	max	n_unique
id	0	1.00	111705.00	64492.77	111705	1	223409	223409
timestamp	0	1.00	1661135705.37	12645.42	1661135577	1661114079	1661158618	42238
price	39116	0.82	239938.98	7562062.01	135000	1	2147483647	2852
n_rooms	60323	0.73	3.50	0.99	3	2	5	4
floor	72365	0.68	1.82	1.13	2	1	52	22
mq	4034	0.98	158.63	128.68	117	1	999	976
n_bathrooms	14397	0.94	1.59	0.67	1	1	3	3
year_of_construction	10	1.00	1965.13	76.75	1980	1000	2209	389
has_garage	0	1.00	0.18	0.38	0	0	1	2
has_terrace	0	1.00	0.11	0.32	0	0	1	2
has_garden	0	1.00	0.17	0.37	0	0	1	2
has_balcony	0	1.00	0.10	0.30	0	0	1	2
has_fireplace	0	1.00	0.05	0.23	0	0	1	2
has_alarm	0	1.00	0.01	0.10	0	0	1	2
has_air_conditioning	0	1.00	0.30	0.46	0	0	1	2
has_pool	0	1.00	0.02	0.15	0	0	1	2
has_parking	0	1.00	0.02	0.12	0	0	1	2
has_elevator	0	1.00	0.06	0.23	0	0	1	2
is_furnished	0	1.00	0.08	0.27	0	0	1	2

7.2 Convergence of the algorithm and plausability of the imputed dataset

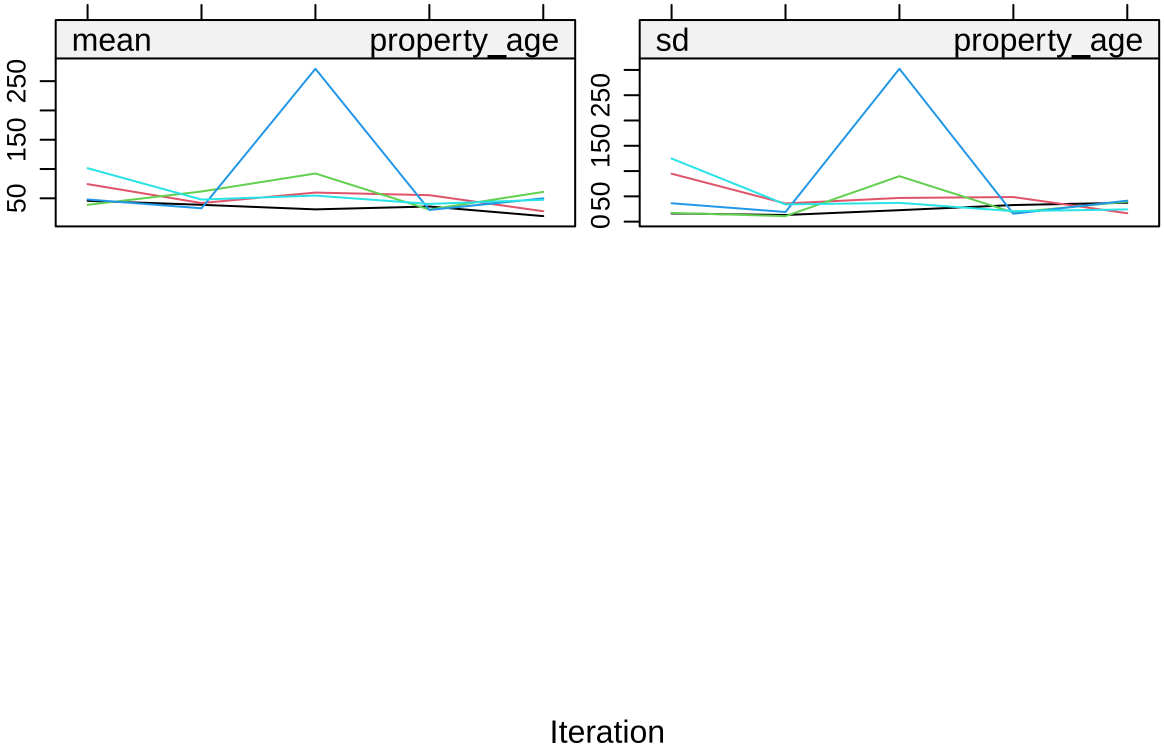
```r
# convergence of the algorithm
plot(imputations)
```



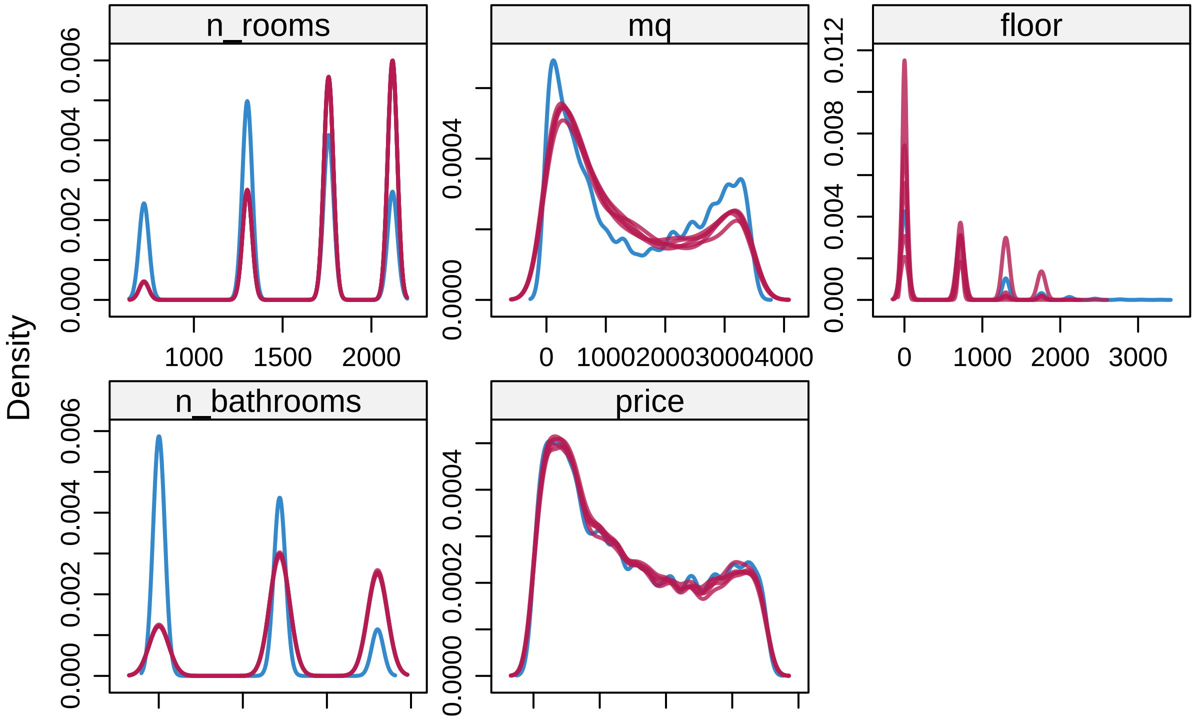
mean
price
184000
187000
sd
price
158000
164000
mean
n_rooms
3.14
3.15
3.16
sd
n_rooms
0.830
0.834
mean
floor
1.5
2.0
2.5
sd
floor
0.5
1.0
1.5
1
2
3
4
5
Iteration



mean
mq
194
198
202
sd
mq
150
155
160
mean
n_bathrooms
2.17
2.19
2.21
sd
n_bathrooms
0.714
0.720
mean
energy_class
9.5
9.7
9.9
sd
energy_class
3.3
3.4
3.5
3.6
1
2
3
4
5
Iteration



```
# plausibility of the imputed data
densityplot(imputations, ~n_rooms + mq + floor + n_bathrooms + price, lwd = 2)
```


Density
n_rooms
0.000 0.002 0.004 0.006
1000 1500 2000
mq
0.0000 0.0004
0 1000 2000 3000 4000
floor
0.000 0.004 0.008 0.012
0 1000 2000 3000
n_bathrooms
0.000 0.002 0.004 0.006
0 500 1000 1500
price
0.0000 0.0002 0.0004
0 1000 2000 3000 4000