**Technical report - Tarmo Lindfors - Atte Kokko**

Our plan was to find the best housing investment areas in Finland and make easy to read visualizations mainly for individual Housing investors. We wanted to analyze the prices and rents of different areas of cities in Finland and compare the profits between them. In the end we chose Helsinki, Vantaa, Espoo, Tampere and Turku to be our cites. We thought these would be the top investment areas, and cutting out small cities removes clutter from the future analysis and visualizations. For our tools we used python and its libraries for processing and plots. For collaboration we chose GitHub for easy code sharing and github.io for communicating our result on a web page.

**Data Collection:**

We found our data sets from Tilastokeskus. For house prices and rents for three different apartment sizes. Tilastokeskus gathers the rents for different areas by combining the data of people getting housing benefits from Kela and The Digital and Population Data Services Agency's data on buildings and dwellings. This means that the most expensive rent apartments might not be included in the data, as they likely do not get any benefits from Kela. On the other hand, the house prices are gathered from the Tax Administration's asset transfer tax statements, so the data can be interpreted as reliable as there is no guesswork or approximations needed for either. Sample size is left as the best indicator for poor data quality. In the later parts of preprocessing, we noticed the best option for filling in the missing values would be to use Consumer Price Indices, these were also found from Tilastokeskus.

We downloaded all data sets in csv format for quarterly data separated by postcodes and into 1-,2-,3-bedroom sizes houses and starting from year 2015 onwards as otherwise we would need to dig older data from archives of Tilastokeskus. For the house prices we encountered an issue with the Tilastokeskus export system. It only allows 300 thousand datapoints to be exported at once. Easy fix was to download apartments sizes as separate files and combine them in the preprocessing step. We could've filtered the cities in this step already, but we believe pandas give us the easiest approach as Tilastokeskus requires clicking with mouse for filtering. The file sizes aren't too big either.

CPI was reported as a simple list of monthly indices. Tilastokeskus includes the different components which are used in the gathering of the total CPI index. We used the "Todelliset asumisvuokrat" and "Osakehuoneistot ja kiinteistöt" for whole of Finland. These Tilastokeskus gathers from the datasets we already use in the main analysis. This should be a fine estimate of the world trend of the evolution of the prices and rents.

**Preprocessing:**

The file in the repository that contains the code used in this part is proju.ipynb.

The collected data from Tilastokeskus was filled with nan values and contained a lot or irrelevant places in the context of the projects goal: housing investment. Even if the rental incomes seem good compared to purchase price of a house in a provincial area, the amount of willing to rent is likely to be low making the investment risky business. Therefore, we made sure to only contain data that was taken from either Turku, Tampere or Helsinki metropolitan area.

The data from Tilastokeskus had a single column with info of postal code, city part and city for each data entry. This column was divided with NumPy into 3 different columns so that data can be processed more easily in the future. With the new city column all data that was not in the before mentioned cities was dropped. This process was done to both house price and rental price data.

The next problem was the vast amount of NaN values in the collected data. Dropping these values would have made the sample size for the experiment unfortunately small, so it was decided to fill the NaN values instead. The difficulty with this process was to fill the NaN values in a convincing way that would take into context the average prices of that time in question and the area's general price range. Another thing to be taken into context here was the fact that the data was also divided by the number of rooms in each entry.

We ended up filling the NaN values by iterating through the data for each distinct postal code keeping the already seen postal codes in a set. For each postal code, the data had to be iterated through for each room value as well (single, two and three+ bedrooms). After collecting one of these collections, where the room count and the postal code is the same, the missing NaN values could be filled, by standardizing the latest non NaN value into its corresponding value of 2015. This was done with the quarterly CPI index percentage, who's standard 100% value corresponded to 2015 quarter 1 value. Because all of these index values were available, the standardized value could be turned into the value that corresponded to the missing year's value. The code was also optimized so that it will likely use different available values when filling in the missing values, to avoid a possible bias towards one year's value.

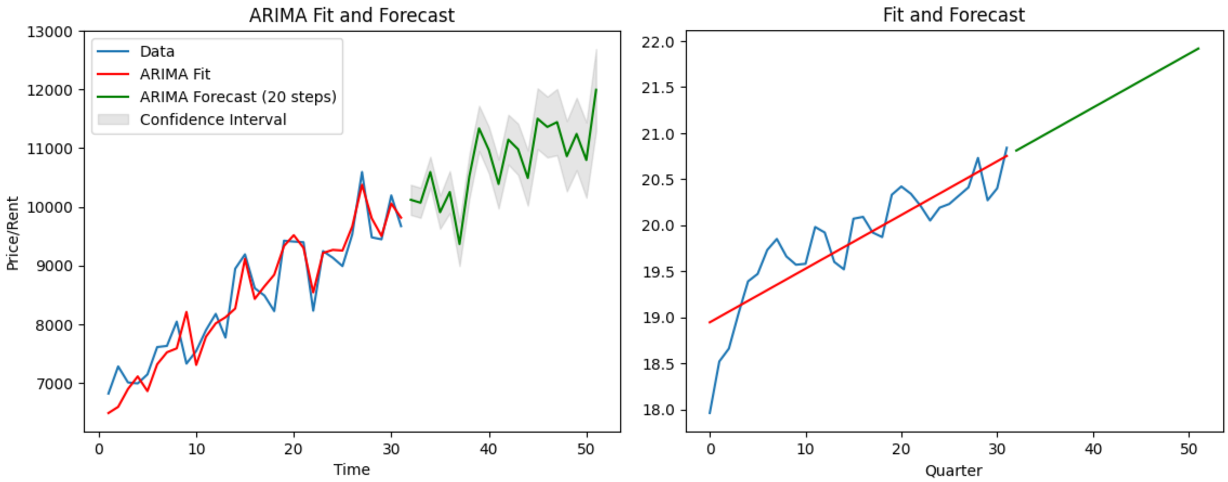
These now filled data frames, both prices and rentals, were then combined, so that the same valued room counts, and postal code areas and cities were matched, to make sure that the correct prices were matched with the correct rentals. It is important to remember that in the case of all values missing in some postal code and room count collections, the values could not be filled. There was not a single year's value available, from which to take the price to be standardized. This led to the problem that some areas had their rentals available, but the prices were missing, or vice versa. These areas were unfortunately dropped from the final data, as the missing values would have had to be made up. There also were some cases of the price data containing some area and the rental data missing it, so those were also dropped. Overall, the data shrunk from around 16 000 rows to 11 000 rows.

In this combined data (in repository with name merged_data) the value of price compared to rent is also calculated, which can be imagined as the number of months required to get the initial investments price in return from the raw rental income. In other words, the lower this new value in the difference column, the higher the raw return is compared to initial investment. This value is used to draw the graphs in visualization part, although it is turned into 1/difference so that the lower payback times seem larger than the higher payback times. Compared to the canvas made beforehand, the amount of preprocessing ended up taking a large part of the hours spent working on the project.

**Machine learning:**

The code used in this section can be found in repository under file arima.ipynb

Our plan was to predict the future development of the housing rents and prices, to see the long-term profit opportunities these different areas have. First, we tried to use ARIMA as this was suggested by the assistant, but in the end, we found out that this is not the best option for us. Implementing the ARIMA, we used python package statsmodels, and created a class which could be used to fit prediction and plot the result separately for every postcode area.



Picture 1: Arima fit on the left and slope fit on the right

ARIMA was able to find the cycles in our data. But to predict the large changes in prices or rents, we would require data about surroundings or politics. Or perhaps more historical data could help ARIMA itself, maybe double our current 9 years. As currently ARIMA would have predicted a large jump to happen again if there had been one in our data (like the jump on picture 1 on the right). Because of these large jumps, we also were not able to find fit parameters that would work on all the different areas. So fine tuning these would have required too much manpower.

But when it works it makes actually pretty good guess of the future (left side of picture 1), but because of the variance included by design in ARIMA, our final prediction would include random chance. This also conflicts with our goal of giving a simple visualization of the potential of different areas.

We decided to focus on the trend of the prices and rents, which seemed to be mainly constant except for those large jumps in some areas. And even then, the slope follows well the trend of the area (picture 1 right side). And by using the already implemented class, it was easy to change the ARIMA to slope fit.

We decided to predict about 5 years into the future starting with 2023 (2023 is still missing in the data of Tilastokeskus). This seemed like a nice compromise for a little look into the future, and hopefully not to be completely wrong. These results were written into another csv in the same format as the merged_data.csv for easy visualization in the future.

**Visuals:**

The code used in this section can be found in repository under file visuals.ipynb

The main goal of visualization was to give an easy-to-read format explaining the findings to the product receiver. We ended up making three different graph templates, each taking a unique approach to the data.

The best way to show the customer the best possible investment opportunities was to create a bar chart. The size of each bar in the chart depicts the before mentioned difference values inversion. The code used to create these charts allows a freely chosen number of bars to be created, but scaling the texts displaying info on each bar must be scaled separately. The code can also display the worst ones, if user so decides.

The bar chart does not display the bigger picture well. It can display around 20 rows of data at once but becomes quickly hard to read. To display information on a bigger scale we used a scatterplot, where y axis corresponds to price and x axis to rental income. Each dot in these scatterplots depicts one postal code area. There are 2 variants of this plot, one with dots color-coded by the count of rooms, and another where the dots are color-coded by city the postal code is from. With these 2 scatterplots we can make it clear to the customer, what areas are good/bad in general and what type of housing investment the end user should look for (mostly single bedroom apartments)

**Website:**

The static website is made so that the end user has easy access to the data. It is made using html and CSS, no JavaScript. The hosting of the website is run by GitHub pages. The website can be updated in the future, with the updates instantly going live to the end user.

**Reflection:**

Overall, the workload of the project was as expected, and the making process went about the same way as expected. In the original idea there was not as big of an emphasis on the sizes of the houses, but separating those made the data a lot more reliable, as same sized houses are clumped together. We thought it was very educational to see the data science process as the project moved on and it gave a good experience for a possible future work project.

Due to neither of us being familiar with NumPy or other data science related software, the learning process made the project maybe a little shorter than would have been ideal. Now the customer has no easy way to look at the data on some specific year, or get the graphs related to that year, so that would have to be done by us manually (or expect the customer to be familiar with python). If the prototype was advanced more, the next step would definitely be a web application instead of the static website currently in use, although the customer does get the general results of the project already (meaning what areas could be those hidden gems waiting to be purchased and what type of housing). There could also be more aspects taken into context, such as average property tax per area or give the user the option to define these for different areas (if the web app would be implemented).

For the future prediction, more data could make ARIMA better. But as discussed, the random chance, still included by it, would require different kinds of plots for visualization. Or maybe just notifying the user about the areas that have had a big change in prices or rents. What might these large changes mean for the investment potential?