**Curneu**

**Problem Statement-1 Documentation**

**Problem statement-1**

Explore the given brazil house rent data set using EDA techniques visualize the results and build a suitable model to predict the house rent.

**Dataset:**

This dataset contains 10692 properties to rent with 13 different features

city: City where the property is located

area: Property area

rooms: Quantity of rooms

bathroom: Quantity of bathroom

parking spaces: Quantity of parking spaces

floor: Floor

animals: Accept animals?

furniture: Furniture?

hoa (R$): Homeowners association tax

rent amount (R$)sort: Rent amount

property tax: municipal property tax

fire insurance (R$): fire insurance value

total (R$): the sum of all values

**Data cleaning and Pre-processing**

**Data cleaning** is the process of preparing **data** for analysis by removing or modifying **data** that is incorrect, incomplete, irrelevant, duplicated, or improperly formatted. This **data** is usually not necessary or helpful when it comes to analysing **data** because it may hinder the process or provide inaccurate results.

* Check if there is any null values in the dataset
* Replacing categorical columns with numerical values

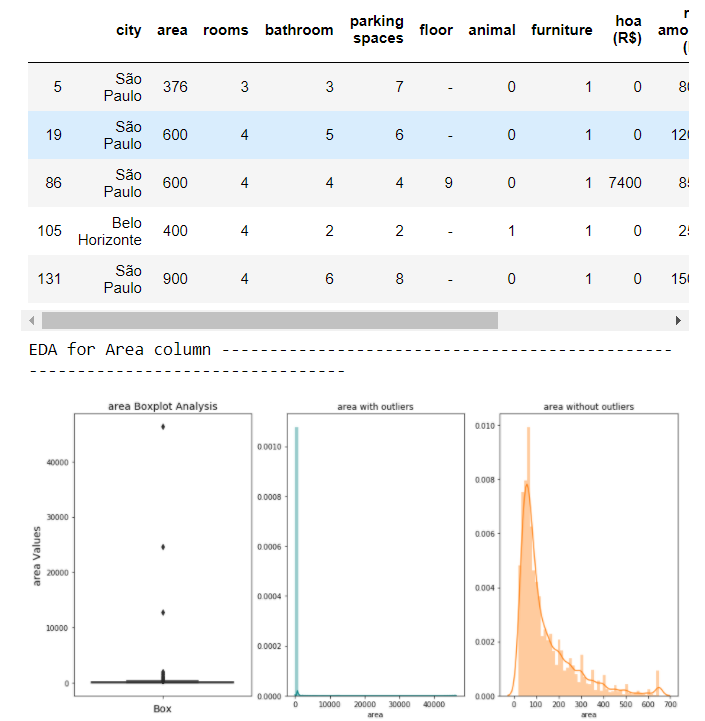
**Exploratory Data Analysis(EDA)**

**Exploratory data analysis (EDA)** is used by data scientists to analyze and investigate data sets and summarize their main characteristics, often employing data visualization methods

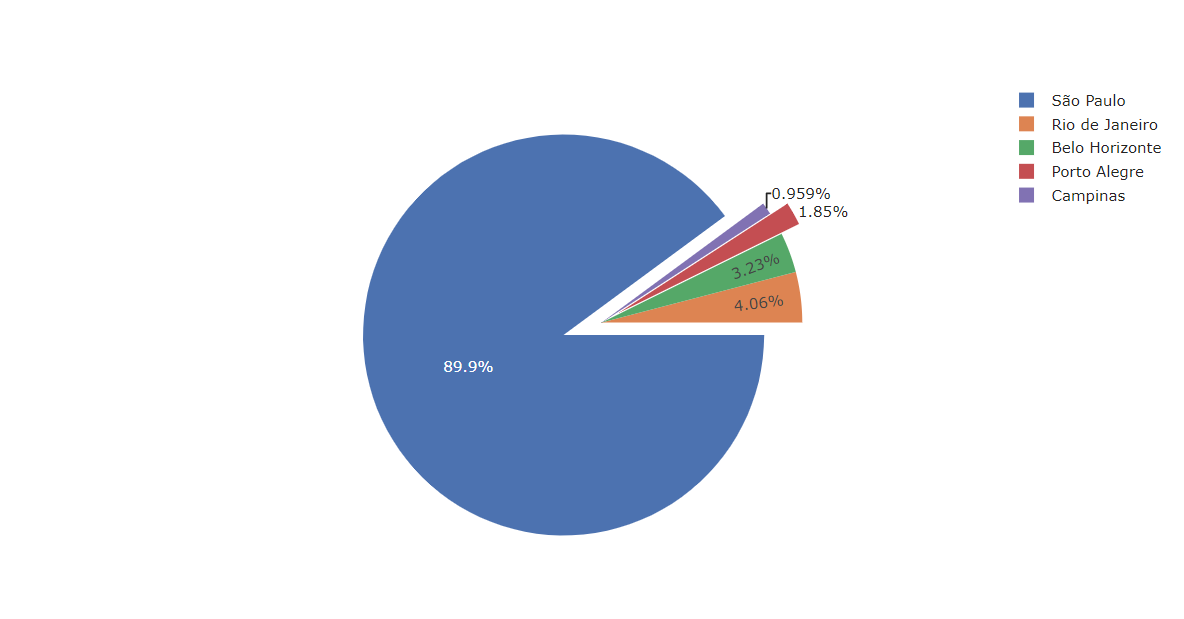
**Univariate Analysis**

**Univariate analysis** is the simplest form of analysing data. “Uni” means “one”, so in other words your data has only one variable. It doesn't deal with causes or relationships (unlike regression ) and it's major purpose is to describe; It takes data, summarizes that data and finds patterns in the data

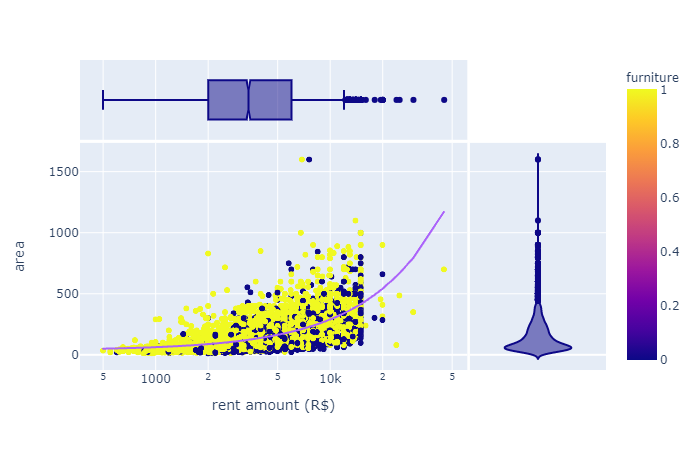
* Constructing a boxplot to each the outliers for each column in the dataset
* Removing the outliers and constructing a new DataFrame
* Distribution chart to check how the data is distributed without outliers



* Checking which city has the city rented houses

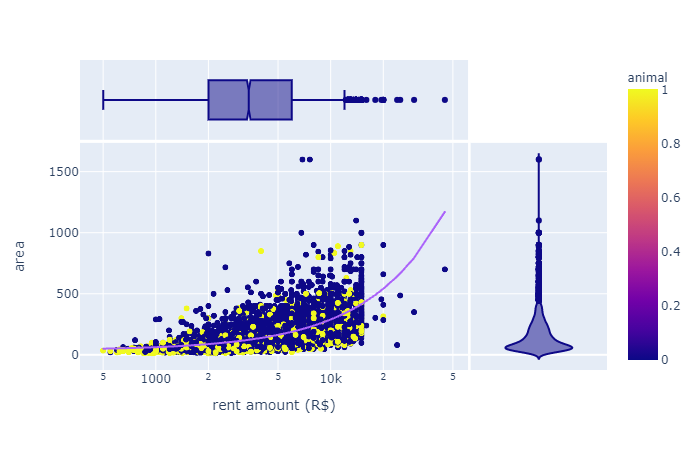


* Checking if furnished feature affects the rent. So, we plot it with rent as x-axis and area as y-axis



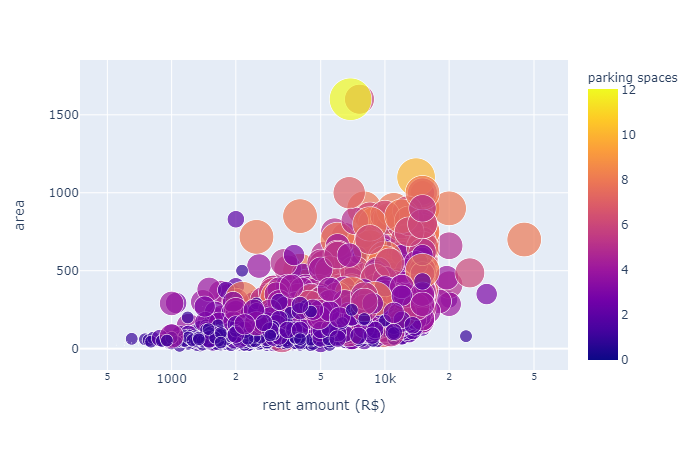
**From the chart we can say that if the house is furnished its rent is high**

* Checking if animal acceptance feature affects the rent.So,we plot it with rent as x-axis and area as y-axis

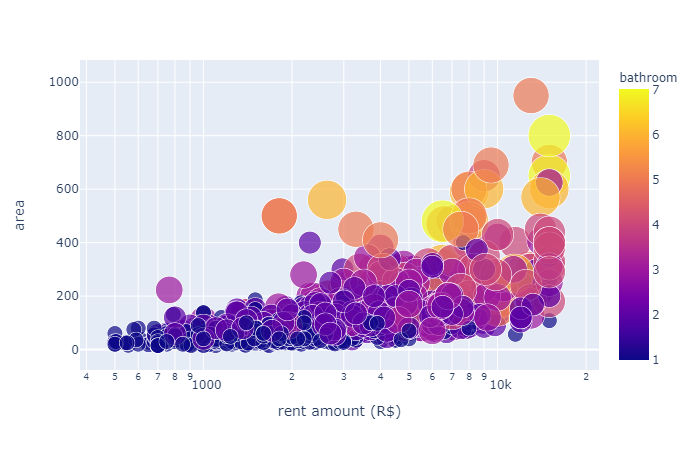


**From the chart we can say that the acceptance of animal doesn’t affect rent much**

Checking if parking space feature affects the rent.So,we plot it with rent as x-axis and area as y-axis

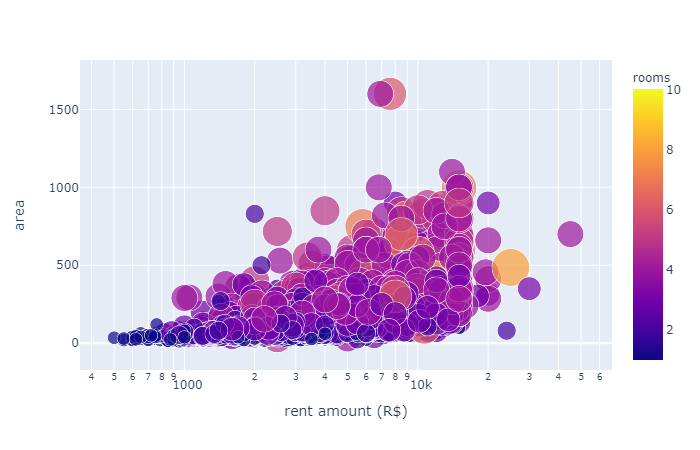


**It is obvious that more parking area more the area needed so obviously rent will increase**

* Checking if no. of bathrooms feature affects the rent.So,we plot it with rent as x-axis and area as y-axis.

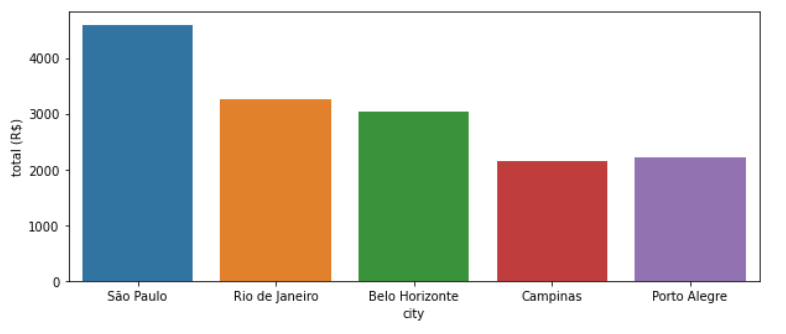
**It is obvious that more no. of bathrooms more the area needed so obviously rent will increase**

* Checking if no. of rooms feature affects the rent.So,we plot it with rent as x-axis and area as y-axis.



**It is obvious that more rooms more the area needed so obviously rent will increase**

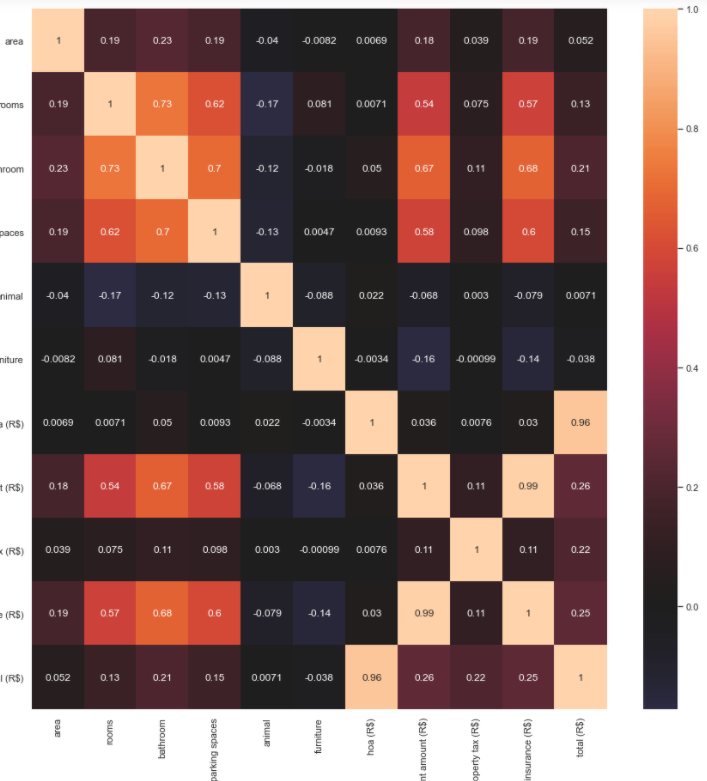
* Plotting which has the highest total



**It is highest is Sao Paulo**

**Constructing Correlation matrix to check the variable relationships**

A **correlation matrix** is a table showing **correlation** coefficients between variables. Each cell in the table shows the **correlation** between two variables. A **correlation matrix** is used to summarize data, as an input into a more advanced analysis, and as a diagnostic for advanced analyses



From the matrix we can see that few variables are highly correlated and few are not.

**Fire insurance is highly correlated where hoa is not.So we remove hoa.**

**Data Splitting:**

The basic idea is to divide the dataset T into two subsets – one subset is used for training while the other subset is left out and the performance of the final model is evaluated on it. The main purpose of cross-validation is to achieve a stable and confident estimate of the model performance

* We split the model into 75% for training and 25% for test

**Data scaling :**

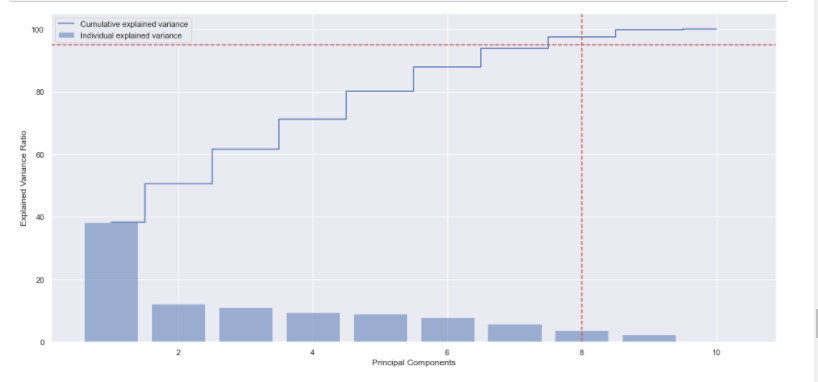
Feature **scaling** (also known as **data** normalization) is the method used to standardize the range of features of **data**. Since, the range of values of **data** may vary widely, it becomes a necessary step in **data** preprocessing while using machine learning algorithms

* We scale the data for all the variables to make their variable values from 0 to 1.

**Principle Component Analysis:**

**Principal component analysis** (**PCA**) is the process of computing the principal components and using them to perform a change of basis on the data, sometimes using only the first few principal components and ignoring the rest.

* We use PCA to find the principle component of the that affects the target variable



**Since the cumulative explained variance falls after 8,we take only 8 components.**

**Model Selection**

**Decision Tree Regressor:**

**Decision tree** builds **regression** or **classification** models in the form of a **tree** structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated **decision tree** is incrementally developed. The final result is a **tree** with **decision** nodes and leaf nodes.

**Random Forest Regressor:**

A **random forest regressor**. A **random forest** is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting

**Linear Regression:**

**Linear regression** is a **linear** approach to modelling the relationship between a scalar response and one or more explanatory variables (also known as dependent and independent variables).

**Support vector machine regressor:**

**Support Vector Machine** can also be used as a **regression** method, maintaining all the main features that characterize the algorithm (maximal margin). The **Support Vector Regression** (SVR) uses the same principles as the **SVM** for classification, with only a few minor differences.

**K-Neighbors Regressor:**

**KNN** algorithm **can** be **used** for both classification and **regression** problems. The **KNN** algorithm **uses** 'feature similarity' to predict the values of any new data points. This means that the new point is assigned a value based on how closely it resembles the points in the training set.

**Lasso Regression**

**Lasso regression** is a type of linear **regression** that uses shrinkage. Shrinkage is where data values are shrunk towards a central point, like the mean. The **lasso** procedure encourages simple, sparse models (i.e. models with fewer parameters).

**Ridge Regression**

**Ridge regression** is a way to create a parsimonious model when the number of predictor variables in a set exceeds the number of observations, or when a data set has multicollinearity (correlations between predictor variables).

**Model Evaluation:**

**Accuracy(r2 score):**

**R2** corresponds to the squared correlation between the observed outcome values and the predicted values by the model. The Higher the R-squared, the better the model

**Conclusion:**

Therefore, Random Forest Regressor has high accuracy score than the other models. So we select it and predict the output.

