DTU

DTU Compute

02450 Introduction to Machine Learning and Data Mining (Spring 2025)

Danish Residential Housing Prices

Regression & Classification

Project 2

Table of Contributions

Section	**Dimitar Ilev** (s236115)	**Christian Hayes Vigerust** (s242161)	**Liza Szalai** (s250208)
Introduction	90%	2.5%	7.5%
Regression, Part a	75%	20%	5%
Regression, Part b	85%	10%	5%
Classification	15%	0%	85%
Discussion	50%	10%	40%

Table of Contents

List of Figures

List of Tables

1 Introduction

This project is a continuation of the work initiated in Project 1. Based on the feedback received, several refinements were made to the correlation matrix, heatmap visualization, and the PCA (Principal Component Analysis) results. These updated analyses can be found in Appendix A.1.

The goal of Project 2 is to successfully apply the supervised learning methods for regression and classification. To do so, a subset of the Danish housing dataset [1] was selected to reduce computational complexity. Specifically, a sample of 1,000 observations was extracted from the full dataset of approximately 1.5 million records. This subset was used for training and evaluating the models developed for both regression and classification tasks.

The theoretical and computational foundation for the methods applied in this project is primarily drawn from the lecture notes by Herlau, Schmidt, and Mørup [2]. In particular, algorithms 5 and 6 were used to implement cross-validation, while the statistical evaluation techniques were guided by the methodology outlined in Method Section 11.4 from the same source.

2 Regression

Although the primary goal of regression models is to predict the **purchase price**, this section focuses on analyzing how the captured variance of the model is influenced by the introduction of bias, varying feature weights, and regularization. The analysis draws on four key categories of input features: transaction details, such as sale conditions; property characteristics, including square meters and number of rooms; location attributes, such as ZIP code; and economic indicators, such as interest and inflation rates. As all selected features are numerical, there is no need for one-hot encoding at this stage of the analysis.

2.1 Linear Regression (Part A)

The features were standardized using the `StandardScaler` from `scikit-learn`, which transforms each feature x_i to z_i with zero mean and unit variance. Where μ_i is the sample mean and σ_i is its standard deviation:

$$z_i = \frac{x_i - \mu_i}{\sigma_i} \tag{1}$$

All predictors were treated as conditionally independent, given the target:

$$p(x_1, x_2, \ldots, x_n|y) = \prod_{i=1}^{n} p(x_i|y) \tag{2}$$

The linear relationship between features and the price was modeled as:

$$price = \omega_0 + \sum_{i=1}^{n} \omega_i x_i \tag{3}$$

Regularization via Cross-Validation

10-Fold cross-validation was implemented to select the model with the lowest test error following algorithm 5 for model selection. The regularization factor (λ) was chosen to be in an array interval of 50 steps between 10^{-1} and 10^{6}. The results can be seen in Figure 1. The graph on the left shows how the coefficient values change with the strength of regularization. The right graph shows the mean squared error (MSE) as a function of regularization.

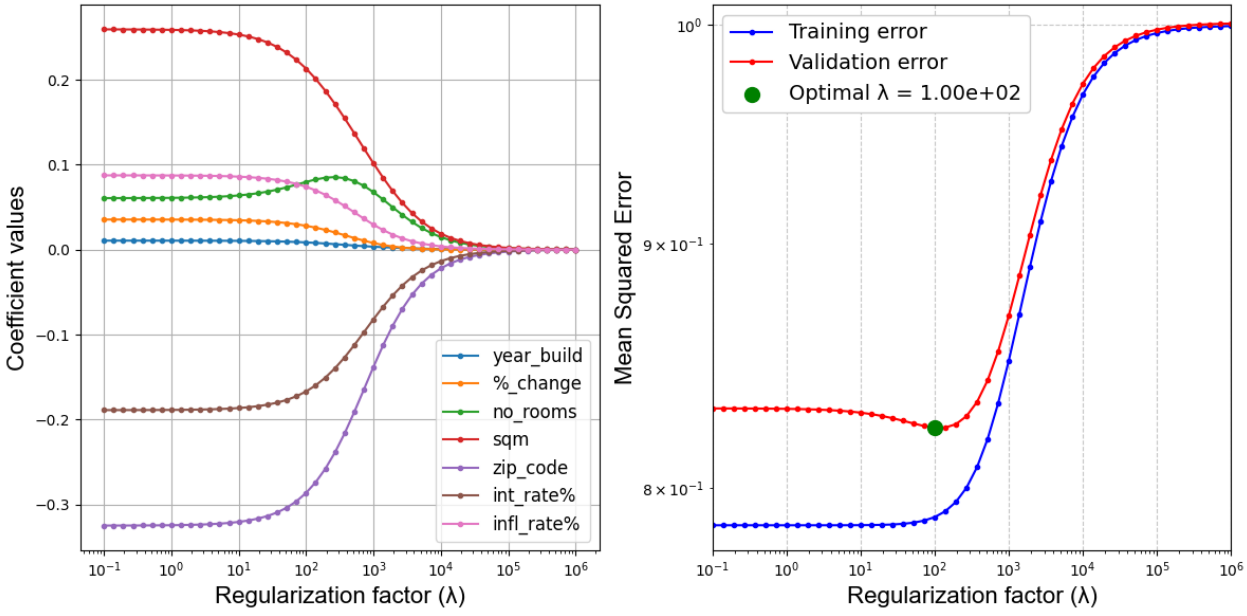

Figure 1: Ridge regression using λ generalization

It becomes evident that for lower λ, the coefficients capture a lot of variance, leading to overfitting on the training set and resulting in a high validation error. The optimal λ=100 gave the lowest validation error. It is marked with a green dot at the bottom of the dip of the validation curve. For the optimal regularization factor, features such as the year of build seemingly have a relatively low influence on the model, while zip code, inflation rate, and square meters have the highest influence on the purchase price.

Model	Training MSE	Test MSE
Baseline (mean prediction)	1.00	1.02
Ridge Regression (λ = 100)	0.80	0.86

Table 1: MSE ridge regularization comparison vs baseline

Table 1 shows a comparison between the mean baseline model and the regularized linear model. Using an optimal λ will lead to a much smaller error in both the training and testing of the linear model. Drastically reducing the MSE from 1 to 0.8 on the training and from 1.02 to 0.86 on the testing set.

Linear Model Output and Attribute Effects

The *generalization error* represents the average *MSE* across all K validation folds and can be seen as the red line in the graph to the right in Figure 1. It accounts for the error between true purchase prices in the validation set and purchase prices predicted by a simple linear model with the corresponding observations as input. The simple linear model uses a weight vector **W** to transform an observation of multiple attributes of the input vector $\mathbf{x_i}$, into a single output y_i. This is done according to equation 4:

$$f(\mathbf{x_i}, \mathbf{w}) = \tilde{\mathbf{x}}_i^\top \mathbf{w} = y_i, \tag{4}$$

The benefit of ridge regularization is that it reveals the optimal weights **w** with the least amount of overfitting. The weights are found by minimizing the corresponding cost function of rigid regularization, using λ as the regularization factor:

$$E_\lambda(\mathbf{w}, w_0) = \left\| \mathbf{y} - w_0\mathbf{1} - \hat{\mathbf{X}}\mathbf{w} \right\|^2 + \lambda \left\| \mathbf{w} \right\|^2, \quad \lambda \geq 0. \tag{5}$$

The coefficient values on the left graph of Figure 1 correspond to the components of **w** in equation 4 for a given regularization factor. The effect of each attribute is determined by its weight, the variance between the weights is reduced with a higher regularization factor.

Feature	Coefficient
year_build	0.0083
% change	0.0283
no_rooms	0.0808
sqm	0.2149
zip_code	-0.2902
nom_interest_rate%	-0.1690
dk_ann_infl_rate%	0.0755

Table 2: Feature coefficients of ridge regression

Table 2 contains the attribute weights, associated with λ=100. In light of Equation 4, a larger absolute value of a coefficient ω for a given attribute $\mathbf{x_i}$ indicates a stronger influence of that feature on the predicted purchase price. For instance, the variable *sqm* (square meters) exhibits a strong relationship with property value, as larger homes typically command higher prices. As anticipated, features with more extreme coefficient values tend to align with those that demonstrated stronger positive or negative correlations with the purchase price in Project 1. This observation is consistent with the findings presented in the correlation matrix shown in Appendix A.1.

2.2 Model Comparison (Part B)

Two-level cross-validation

Table 3 presents the results of the two-level cross-validation performed on *K1 = K2 = 10* times following algorithm 6. A single-layer layer artificial-neural network was chosen as method two, besides the linear model from section 2 and the Baseline model capturing the mean of *y*. Both ReLU() and Tanh() was tried out as the non-linear activation function. ReLU() was chosen for the final analysis, however, the difference in performance was small. For the tuning of the hyperparameter, the complexity-controlling parameter for the ANN was the number of hidden units (h), including 1, 6, 8 and 10, and for the ridge model a λ array of 50 values between 10^{-1} and 10^{6}.

Outer fold	**ANN**		**Ridge Regression**		**Baseline**
i	h_i^*	E_i^{test}	λ_i^*	E_i^{test}	E_i^{test}
1	6	0.530	100.000	0.580	0.560
2	8	1.100	100.000	0.990	1.260
3	10	0.910	100.000	0.990	1.040
4	6	0.820	100.000	0.790	0.980
5	10	1.090	100.000	1.130	1.620
6	8	0.520	37.276	0.790	0.860
7	8	0.830	26.827	0.920	1.210
8	10	0.890	100.000	1.010	1.210
9	10	0.610	100.000	0.580	0.720
10	10	0.440	100.000	0.520	0.570
Mean	8.6	0.770	86.410	0.830	1.000

Table 3: Regression Two-level cross-validation comparison

At glance the table presents the results obtained for the 10 outer folds in the cross-validation. The mode of hidden units is 10, with an average of around 9 to 10 units and an average test error of 0.77. Making it the best performing model of the three. The mode of λ is 100 the same value estimated in section 2.1, with only two folds tested on lesser regularization values. The model has the second best performance with a mean test error of 0.83. The baseline model has the worst performance with a mean error of 1.0. This is not surprising as the model only accouns for the avearge of the training data, and hence fails to capture significant variance on the testing set.

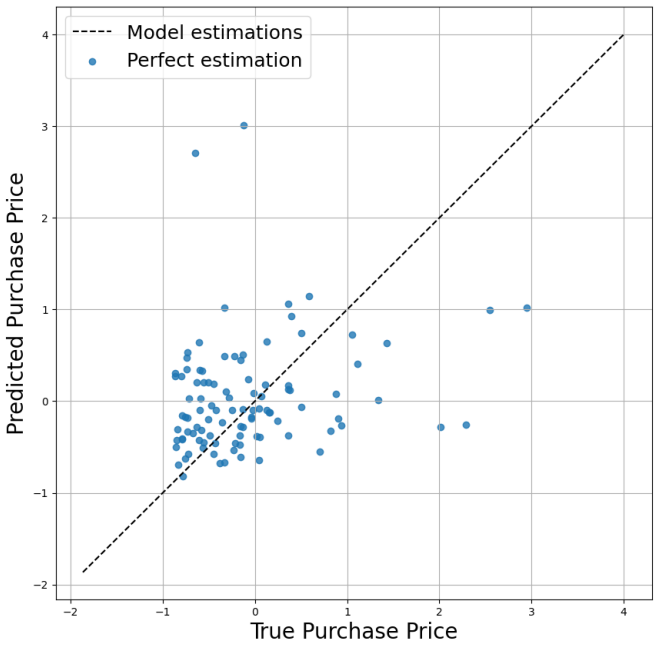

Figure 2: ANN true vs predicted price

Figure 2 showcases the predictions made by the artificial neural network compared to the actual values. It is evident that the model overshoots and undershoots a couple of the predictions, explaining the error rate of 0.77.

Statistical Evaluation

Table 4 presents the pairwise statistical evaluation of the three models using Setup II as described in Section 11.4. This method provides a more realistic estimate of generalization performance, as it evaluates each model across different resampled training sets, rather than reusing the same folds for model selection and error estimation as in Setup I. This mitigates overly optimistic performance estimates and more closely simulates real-world model deployment.

Based on the results of the statistical comparison, all three null hypotheses were rejected with *p*-values below 0.001, indicating that the observed differences in test error are statistically significant. The artificial neural network (ANN) model significantly outperforms the

H_0	p value	Lower CI	Upper CI	Result
$E^{\text{test}}_{\text{baseline}} - E^{\text{test}}_{\text{linear}} = 0$	<0.001	0.1785	0.1925	H_0 rejected
$E^{\text{test}}_{\text{ANN}} - E^{\text{test}}_{\text{linear}} = 0$	<0.001	-0.3298	-0.2089	H_0 rejected
$E^{\text{test}}_{\text{baseline}} - E^{\text{test}}_{\text{ANN}} = 0$	<0.001	0.3914	0.5183	H_0 rejected

Table 4: Statistical evaluation regression models

baseline model, with a 95% confidence interval for the difference in test error ranging from 0.3914 to 0.5183. This indicates that the ANN's advantage is not only consistent across folds but also statistically robust. Similarly, the ridge regression model shows a statistically significant improvement over the baseline, with a confidence interval between 0.1785 and 0.1925.

These findings suggest that both regularization (in ridge regression) and non-linear modeling (in the ANN) enhance the model's ability to capture meaningful variance in the test data. However, the superior performance of the ANN highlights the value of incorporating non-linear transformations. Therefore, for applications where predictive performance is critical, the use of non-linear models like ANNs is strongly recommended over simpler linear approaches.

3 Classification

In the classification problem, the target attribute was the house type, which is a multiclass classification problem with 5 different classes (villa, apartment, summerhouse, farm, or townhouse). Since this attribute is nominal, one-out-of-K-coding was applied. Furthermore, due to significant class imbalance, from the original dataset (1.5 million observations), firstly from each class, the same number of observations (70000) was randomly selected, and from this balanced set of 35000 observations was the subset of 1000 observations created to reduce computational needs. The classifiers use 7 attributes, the year the building was built, the purchase price, % change between the offer and the purchase, the number of rooms, the size of the house (sqm), the price/sqm, and the zip code.

3.1 Model Comparison & Two-level cross-validation

Three models were used and tested for the classification problem, a logistic regression, an artifical neural network, and a baseline model. Two-level cross-validation was used with K1=K2=5 outer and inner folds to decrease runtime. The cross-validation was imple-

mented using algorithm 6 [2].

The baseline model was set to be the most common class in the training set of the outer fold. The complexity-controlling parameter for the logistic regression model was λ, and for the ANN, the number of hidden units. Based on several test runs, the λ values were chosen as an array of 20 logarithmically spaced values in the range of $[10^{-5}, 10^{3}]$ and for $h = [1, 5, 7, 9]$, and ReLU was used as the activation function for the model. The results of applying the classifiers in the last outer fold can be seen in Figure 3.

The error measure used for the model comparison was the error rate of the classifiers, calculated as:

$$E = \frac{|Misclassified\ observations|}{|Test\ set|} \tag{6}$$

The optimal values of the complexity-controlling parameter can be seen in Table 5, as well as the corresponding estimated generalization error rates that were obtained by evaluating the best models on the test sets of the outer folds.

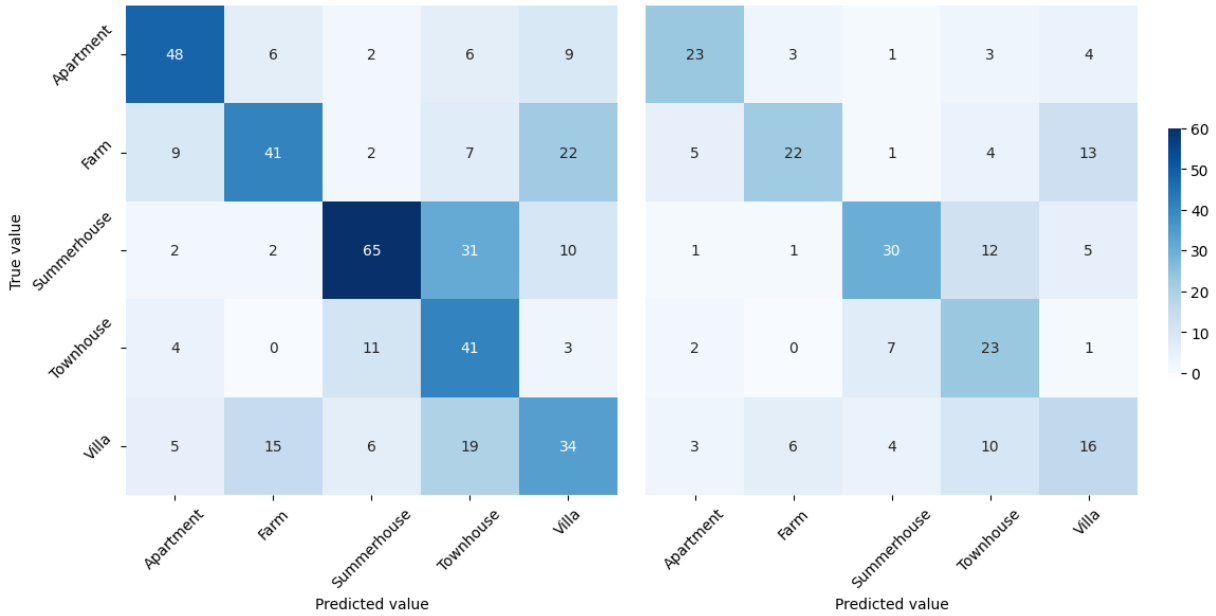

Figure 3: Number of the estimated and true labels after applying a logistic regression (left) and an ANN (right) model from the last outer fold.

These results indicate that the ANN and logistic regression models perform similarly on this dataset and both provide better performance than the baseline model. Since there are 5 classes in the dataset, the average error rate of 0.782 of the baseline model seems reasonable, since given a balanced dataset, the probability of a random guess being the correct class for an observation would be $\frac{1}{5} = 0.2$.

Outer fold	ANN		Logistic regression		Baseline
i	h_i^*	E_i^{test}	λ	E_i^{test}	E_i^{test}
1	5	0.41	2.976	0.395	0.74
2	5	0.44	2.976	0.445	0.825
3	5	0.395	2.976	0.39	0.765
4	7	0.42	2.976	0.41	0.795
5	9	0.395	2.976	0.425	0.785
Mean		0.412		0.413	0.782

Table 5: Comparison of the 3 classification models after two-level cross-validation

3.2 Statistical Evaluation

Statistical evaluation of the three models was conducted based on **Setup II**. The error rates obtained by applying the models on the test set in the outer folds were compared pairwise, using correlated t-tests with the significance level of $\alpha = 0.05$.

H_0	p value	Lower CI	Upper CI	Result
$E_{\text{ANN}}^{\text{test}} - E_{\text{baseline}}^{\text{test}} = 0$	0.0388	-0.4169	-0.0182	H_0 rejected
$E_{\text{logistic}}^{\text{test}} - E_{\text{baseline}}^{\text{test}} = 0$	0.0368	-0.4181	-0.0219	H_0 rejected
$E_{\text{ANN}}^{\text{test}} - E_{\text{logistic}}^{\text{test}} = 0$	0.1419	-0.0012	0.0060	H_0 not rejected

Table 6: Statistical evaluation of the classifier models

The results of the statistical analyses indicate that the estimated error rates of both the ANN and the logistic regression model are significantly different from the error rates of the baseline model. Given the lower estimated generalization errors observed previously, as well as the negative confidence intervals for the estimated error, it can be assumed that both models have better performance in classifying house types compared to the baseline model. Furthermore, when comparing the logistic regression and the ANN model, no significant difference was observed.

Based on these results, as the ANN and the LR model performed similarly, using the LR would be more efficient as it requires less computational power. However, more tests could be conducted to determine if changing the used attributes or other parameters of the models would improve the performance of either model.

3.3 Logistic Regression

Multinomial logistic regression, similar to multinomial ANN, uses a softmax function that outputs the probability of an observation belonging to a given class. This can be done by reparametrizing the θ parameter of the Bernoulli distribution, resulting in a modified softmax function:

$$\theta = [\frac{e^{x^T w_1}}{1 + \sum_{c=1}^{C-1} e^{x^T w_c}} \cdots \frac{e^{x^T w_{C-1}}}{1 + \sum_{c=1}^{C-1} e^{x^T w_c}} \frac{1}{1 + \sum_{c=1}^{C-1} e^{x^T w_c}}] \tag{7}$$

Where x is the observation and w is a vector of weights. This function outputs the probability for a given observation belonging to each class c, and from these probabilities, the class with the highest probability is chosen.

The logistic regression was trained on a training set with an optimal lambda value ($\lambda = 2.976$). The resulting coefficient values for each feature were extracted and can be seen in Table 7.

Feature	Coefficient
year_build	-0.6820
purchase_price	0.2800
% change	0.2421
no_rooms	-1.4489
sqm	0.0542
zip_code	-0.1846
sqm_price	0.3055

Table 7: Feature coefficients of logistic regression with $\lambda = 2.976$

Since the target variable for the classification and the regression problem was different, the weights of the attributes were also different. In the case of the linear regression model predicting the purchase price, the square meter and zip code attributes had higher absolute weights. On the other hand, for the logistic regression predicting the house type, it was the number of rooms and the year built attributes. These attributes intuitively make sense, as the size of the residence has a very high influence on the purchase price, while the number of rooms and the year they built the house can aid in differentiating between the different types of houses. For example, apartments usually have less rooms compared to farms or villas.

4 Discussion

In the regression analysis 2, we learned that the coefficients and λ regularization parameter affect the ability of the model to generalize when introduced to test data. We successfully managed to find an optimal lambda that significantly improved the model performance. Ridge regression reduced the test MSE by 15.7% vs the baseline, demonstrating its ability to balance the bias-variance tradeoffs. The pairwise t-tests showed that all models are statistically significantly different.

In the classification problem 3, we compared an ANN and a logistic regression model based on how well they predicted the house type given a set of features. Their performances were compared pairwise using correlated t-tests. Both models showed improved performance compared to the baseline model, but had similar error rates when compared to each other, and LR might be the ideal choice concerning the computational power.

4.1 Previous Data Analysis

A 2024 article [3] by Lystbæk and Srirajan investigated the sale prices of Danish residential housing. They presented a regression model to predict sales prices using extreme gradient boosting. However, they used a feature importance estimation beforehand and determined that the location, building area and construction year influenced< the prices the most. This slightly differs from the feature importances we observed, but it is also possible that this was because they were not considering the same set of features as we did. Furthermore, they also observed a similar tendency as the regression models presented here, that for higher prices, the models tend to overestimate the price. They explained this phenomenon with the model having more data to train on in the lower end of the price spectrum, and it is more spread out in the upper end.

5 Collaboration

Discussions and collaborations were the key elements to the development of Project 2. *Open AI* was used as a sparing partner for code optimization and suggestion of possible changes. Especially for the improved look of the individual plots. Additionally, *Grammarly* has been used to correct grammatical errors made in the writing process of this document. The main sources for troubleshooting have been Google such as *StackOverflow*, *Scikit-learn* and similar.

Bibliography

[1] Martin Frederiksen. *Danish Residential Housing Prices 1992-2024*. Kaggle. Accessed: February 22, 2025. 2024. URL: https://www.kaggle.com/datasets/martinfrederiksen/danish-residential-housing-prices-1992-2024/data.

[2] Tue Herlau, Mikkel N. Schmidt, and Morten Mørup. *Introduction to Machine Learning and Data Mining*. Lecture notes, Technical University of Denmark. Version 1.0. Version for print. Accessed: April 9, 2025. 2023. URL: https://www2.compute.dtu.dk/~morup/MLDM2023/.

[3] Michael Sahl Lystbæk and Tharsika Pakeerathan Srirajan. "Machine Learning-Based Feature Mapping for Enhanced Understanding of the Housing Market". In: *International Conference on Engineering Applications of Neural Networks*. Springer. 2024, pp. 530–543.

A Appendix

A.1 Project 1 updated

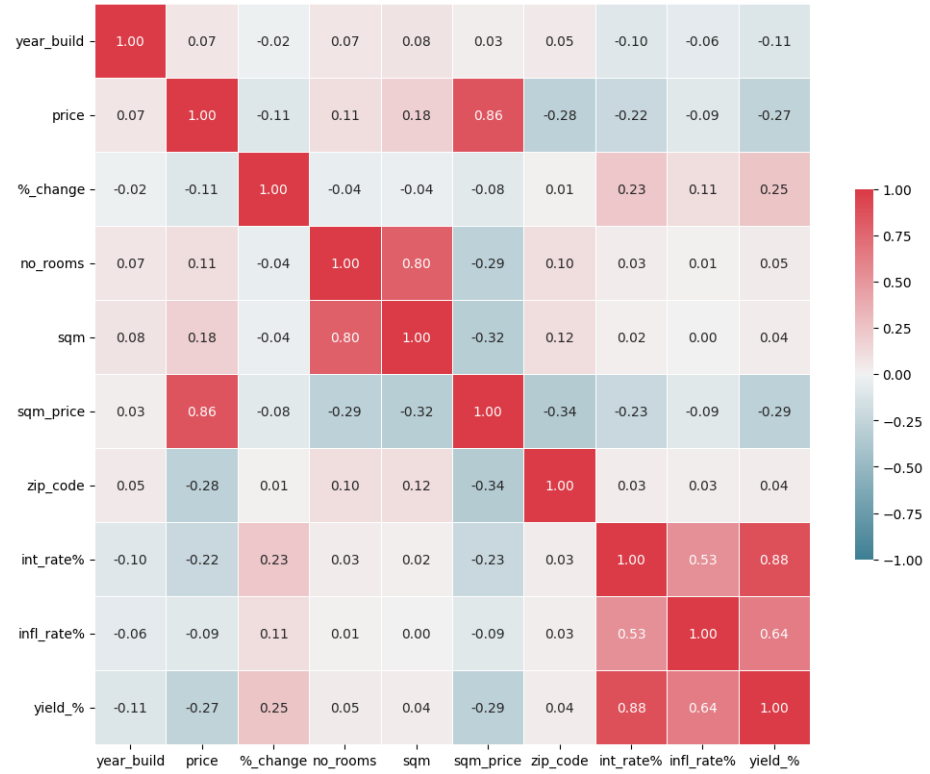

Figure 4: Correlation

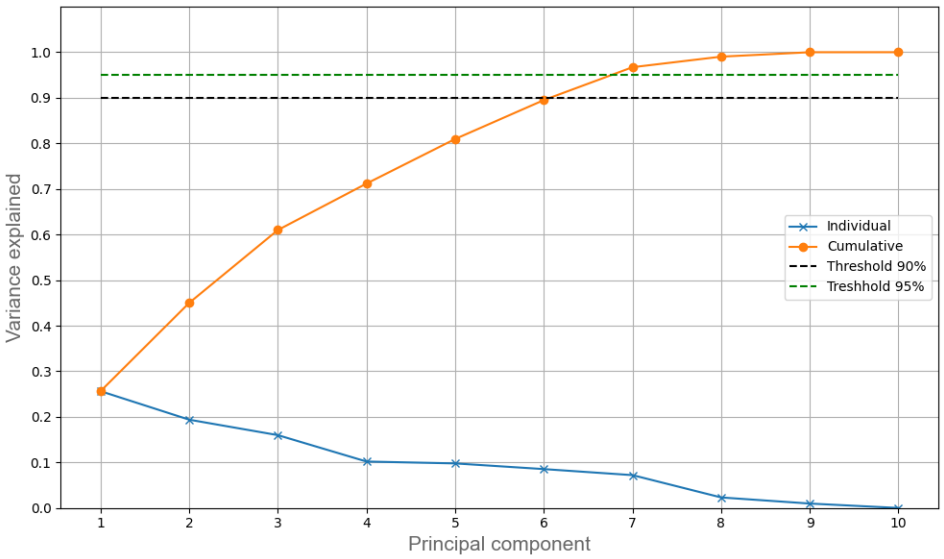

Figure 5: PCA explained variance

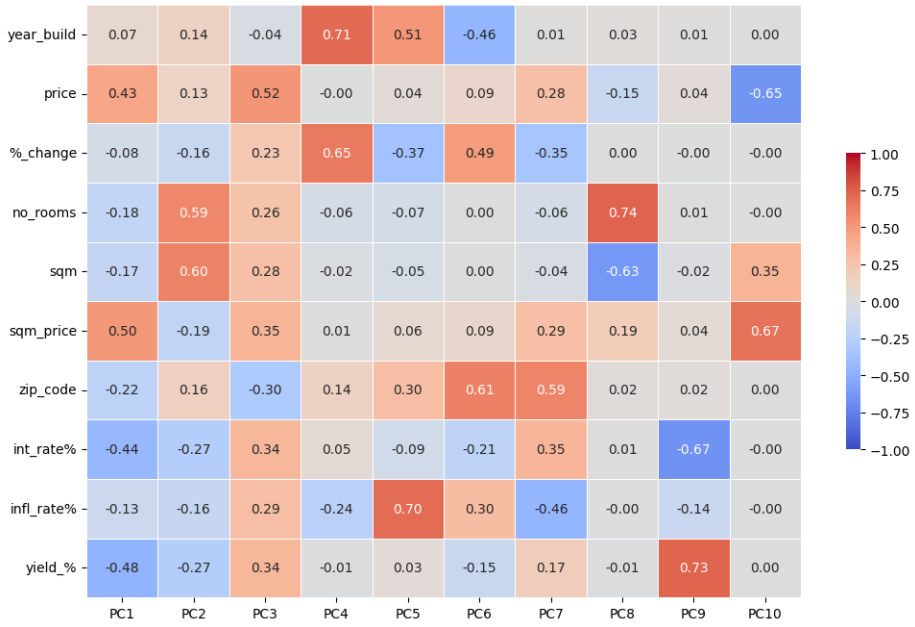

Figure 6: Principal Directions and Attributes

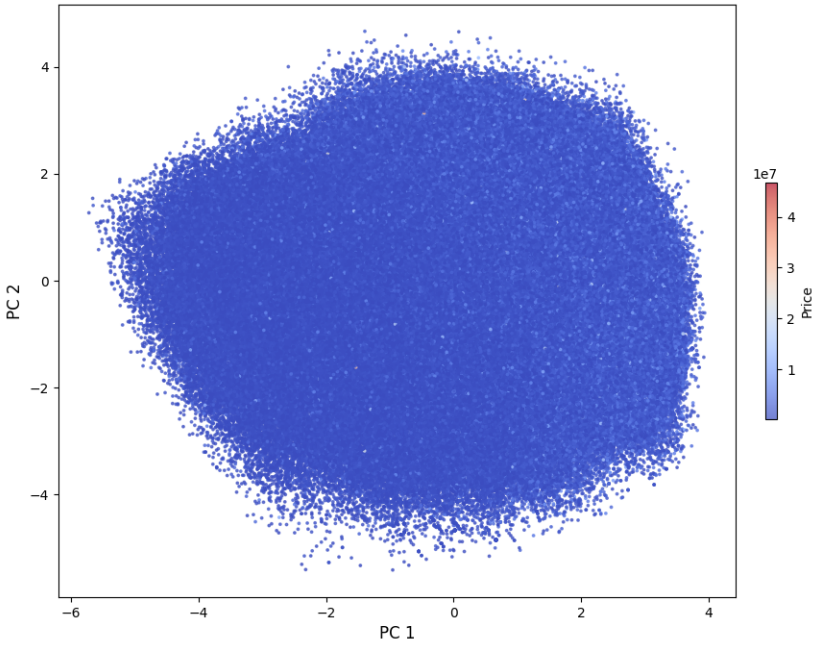

Figure 7: Projection Price on first two PCA

Technical
University of
Denmark

Brovej, Building 118
2800 Kgs. Lyngby
Tlf. 4525 1700

www.byg.dtu.dk