Ridge Regression applied to Housing dataset

Laura Ciurca
Università degli Studi di Milano
Data Science and Economics

Contents

List of Figures

Introduction

1.1 Problem Definition

The aim of the project is defined by the application of ridge regression on the Californian housing dataset, where some of its features are highly correlated, suffering from multicollinearity. Ridge regression is a form of regression, which introduces a regularizer in the ERM to penalize the flexibility of the model and so to increase its stability, making it "stable" when training dataset is modified. Another biased regression technique limiting the multicollinearity problem is principal components regression, where principal components of the explanatory variables are used as regressors. Furthermore, cross validation estimate has been studied to choose the best parameter alpha of ridge regression.

1.2 Brief Algorithm Definition

1.2.1 Ridge Regression

Regression coefficients for linear regression are estimated by:

$$\hat{w} = (S^T S)^{-1} S^T y \tag{1.1}$$

Instead of the sum of squares, Ridge regression adds a parameter α to the identity matrix, which comes from the loss function of the Ridge regression:

$$L = ||Sw - y||^2 + \alpha ||w||^2 = (Sw - y)^T (Sw - y) + (Sw - y) \tag{1.2}$$

By differentiating wrt w and by equating to 0:

$$w = (\alpha I + S^T S)^{-1} S^T y \tag{1.3}$$

While in linear regression, where the expected value of the error term is zero and the variance is constant and finite assuming that the ys are standardized, the estimates result unbiased (Gauss-Markov Theorem), in ridge regression, the amount of bias in the estimator is given by:

$$E(\hat{w} - w) = [(S^T S + \alpha I)^{-1} S^T S - I]w \tag{1.4}$$

Thanks to α, the algorithm's bias can be controlled in return for an increase in efficiency.

1.2.2 Principal Component Regression

Principal Component Regression (PCR) is a regressor that is trained on the training data transformed by Principal Component Analysis (PCA), an unsupervised learning technique used to perform dimensionality reduction, increasing interpretability and minimizing information loss. PCA creates new uncorrelated variables, the so-called principal components, by extracting low dimensional sets of features to capture as much information and in the dataset as possible. In other words, it tries to catch as much variance of data as possible in the new space. Given a dataset x_1,x_2,. . . ,x_m with n-dimensions, PCA allows to project it onto a k-dimensional space with k«n. The procedure for PCA:

1. Standardization of the data:

$$x_{ij} = \frac{x_{ij} - \bar{x}_j}{\sigma_j}, \forall j \tag{1.5}$$

2. Calculate the covariance matrix for the features in the dataset:

$$\Sigma = \frac{1}{m} \sum_{i}^{m} (x_i)(x_i)^T \tag{1.6}$$

3. Calculate eigenvalues and eigenvectors of the covariance matrix:

$$u^T \Sigma = \lambda u \tag{1.7}$$

4. Sort eigenvalues and their corresponding eigenvectors to get the feature vectors, creating the new orthogonal basis for the data.

5. Multiply the transposed feature vectors by the transposed adjusted data, computing the new vector z which comprises $u_k^T x(i)$, being the new representation of the data.

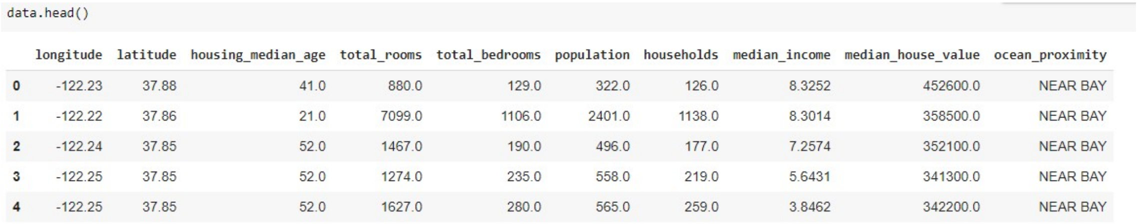
Experimental Evaluation

2.1 Data Cleaning and Pre-processing

The dataset provided presents different attributes that may affect the prices of houses in a specific area. The dataset contains 20640 observations, that have the following 10 features:

1. longitude: A measure of how far west a house is; a higher value is farther west
2. latitude: A measure of how far north a house is; a higher value is farther north. Together with longitude, latitude defines the location of a house in a specific area.
3. housing_median_age: Median age of a house within a block; a lower number is a newer building
4. total_rooms: Total number of rooms within a block
5. total_bedrooms: Total number of bedrooms within a block
6. population: Total number of people residing within a block
7. households: Total number of households, a group of people residing within a home unit, for a block
8. median_income: Median income for households within a block of houses (measured in tens of thousands of US Dollars)
9. median_house_value: Median house value for households within a block (measured in US Dollars)
10. ocean_proximity: Location of houses according to the distance from ocean. The attribute is classified as: <1H OCEAN, NEAR BAY, INLAND, ISLAND, NEAR OCEAN

The “median_house_value” is used as label in order to study the relationship between the presented variables and the corresponding house_price.



```
data.head()
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value	ocean_proximity
0	-122.23	37.88	41.0	880.0	129.0	322.0	126.0	8.3252	452600.0	NEAR BAY
1	-122.22	37.86	21.0	7099.0	1106.0	2401.0	1138.0	8.3014	358500.0	NEAR BAY
2	-122.24	37.85	52.0	1467.0	190.0	496.0	177.0	7.2574	352100.0	NEAR BAY
3	-122.25	37.85	52.0	1274.0	235.0	558.0	219.0	5.6431	341300.0	NEAR BAY
4	-122.25	37.85	52.0	1627.0	280.0	565.0	259.0	3.8462	342200.0	NEAR BAY

Figure 2.1: Californian housing data

Columns are all numerical with the exception of a categorical one, ”ocean_proximity”, which will be handled in Section 2.1.2

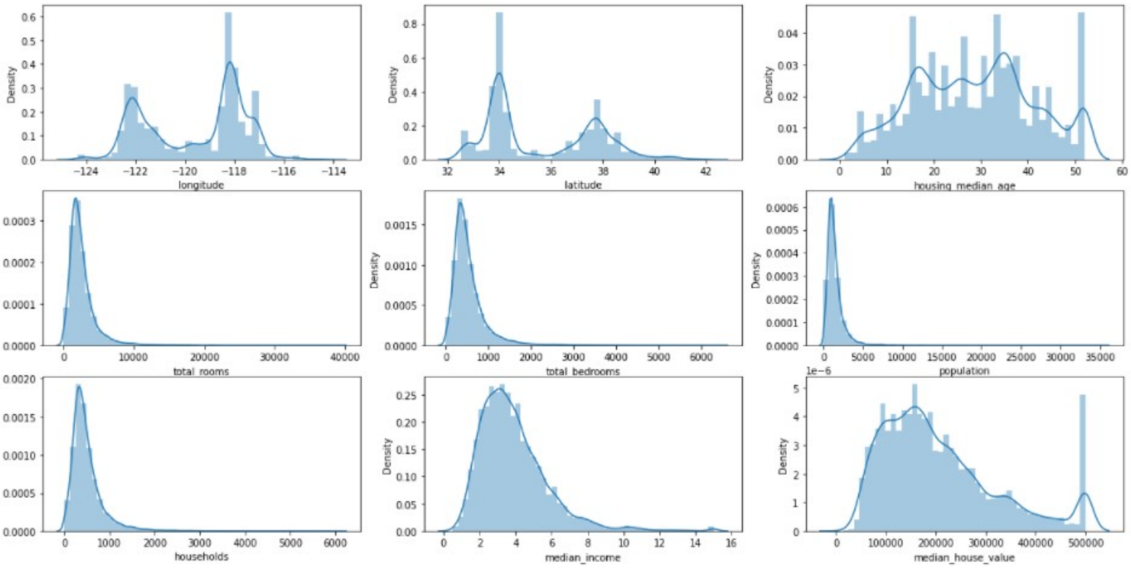
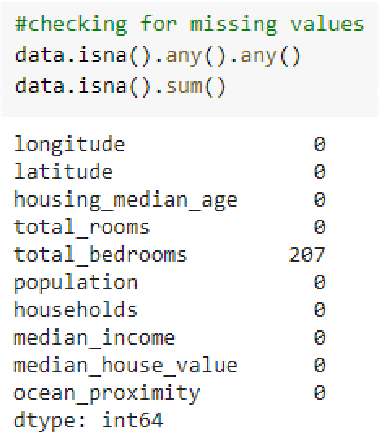


Figure 2.2: Histograms plotting data

From Graphs 2.2, which visualize single numerical variables, it is possible to see that most attributes are heavy on the tails, are skewed and so they will be scaled.
It is also possible to notice that median_house_value presents a high peak for its frequency bigger than 500000 and so it could be an outlier. So these data have been removed.

2.1.1 Missing values

By checking for missing values, it results that only total_bedrooms has non-null values. The attribute “total_bedrooms” contains 207 missing values, that cover a small portion of the dataset and so they have been dropped.



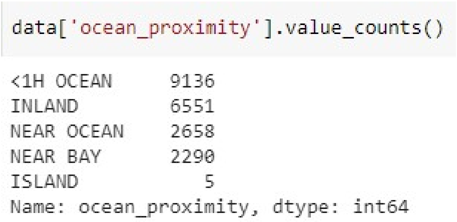
```
#checking for missing values
data.isna().any().any()
data.isna().sum()
```

```
longitude                0
latitude                 0
housing_median_age       0
total_rooms              0
total_bedrooms         207
population               0
households               0
median_income            0
median_house_value       0
ocean_proximity          0
dtype: int64
```

Figure 2.3: NaN values in the dataset

2.1.2 Categorical variables

The variable “ocean_proximity” is categorical, containing five values,<1H OCEAN, INLAND, NEAR OCEAN, NEAR BAY and ISLAND, which has just 5 number of records in the dataset.



```
data['ocean_proximity'].value_counts()
```

```
<1H OCEAN     9136
INLAND        6551
NEAR OCEAN    2658
NEAR BAY      2290
ISLAND           5
Name: ocean_proximity, dtype: int64
```

Figure 2.4: Ocean_proximity values

By looking at the count of each category and at the boxplot in Fig.2.5, it is possible to notice that INLAND presents the highest number of records, but has significantly lower prices with rispect to the others. The exact opposite happens for ISLAND. It makes sense that many people live inland where prices are more affordable that on islands. The distribution in the other features do not change so much.

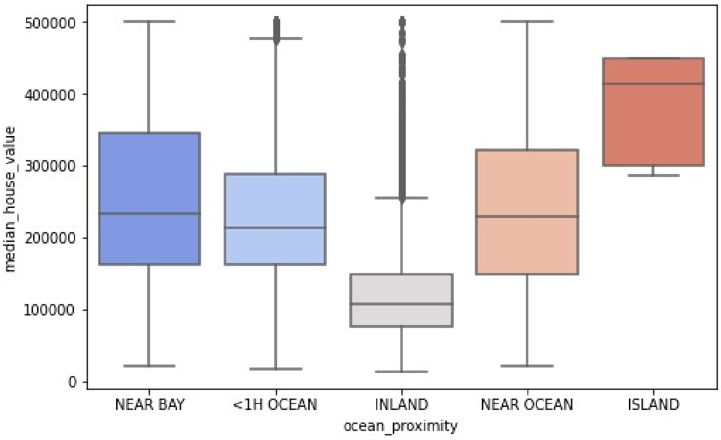
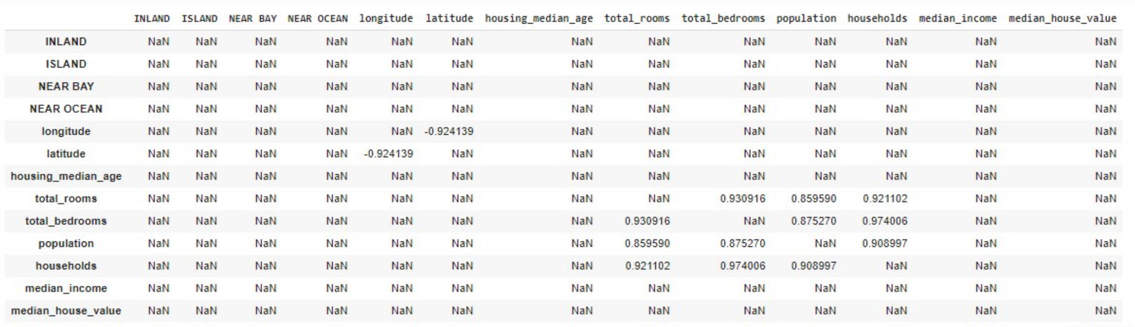


Figure 2.5: Boxplot for ocean_proximity

These categorical values have been replaced with dummy variables. One of the categorical variables has been dropped to avoid the ”dummy variable trap” when running ridge regression.



	INLAND	ISLAND	NEAR BAY	NEAR OCEAN	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
INLAND	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
ISLAND	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NEAR BAY	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NEAR OCEAN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
longitude	NaN	NaN	NaN	NaN	NaN	-0.924139	NaN	NaN	NaN	NaN	NaN	NaN	NaN
latitude	NaN	NaN	NaN	NaN	-0.924139	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
housing_median_age	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
total_rooms	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.930916	0.859590	0.921102	NaN	NaN
total_bedrooms	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.930916	NaN	0.875270	0.974006	NaN	NaN
population	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.859590	0.875270	NaN	0.908997	NaN	NaN
households	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.921102	0.974006	0.908997	NaN	NaN	NaN
median_income	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
median_house_value	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

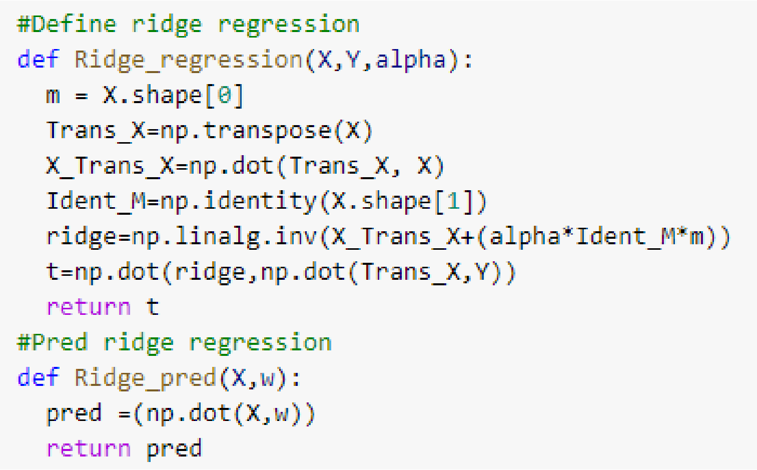
Figure 2.6: Correlation matrix

Multicollinearity is the condition in which high intercorrelations occur among different independent variables. Through the covariance matrix, it is possible to see that some features are highly correlated one to each other. As one can see in Fig.2.6, “totalrooms”, “totalbedrooms”, “population” and “households” are highly correlated. “Totalbedrooms” and “total_rooms” show a correlation of 93%, due to the fact that normally houses with a lot of rooms may be characterized by a lot of bedrooms too. “Population” and “households” present a 91% correlation, since population include all the people within a block, while households could be considered as its subcategory, containing the number of people within a home unit for a specific block. Furthermore, “total_rooms” and “total_bedrooms” are highly correlated to “population” and

“households”, since people living in a certain block or home unit are proportional to the number of rooms and bedrooms of that specific area. The other highly correlated variables are “longitude” and “latitude”, since they are the geographical reference for the house distribution.
In order to have less correlated variables, it is possible to combine them, create new variables or drop some of them. Since houselholds is a sort of ”subcategory” of population and total_bedrooms a ”subcategory” of total_rooms, let’s not consider them.

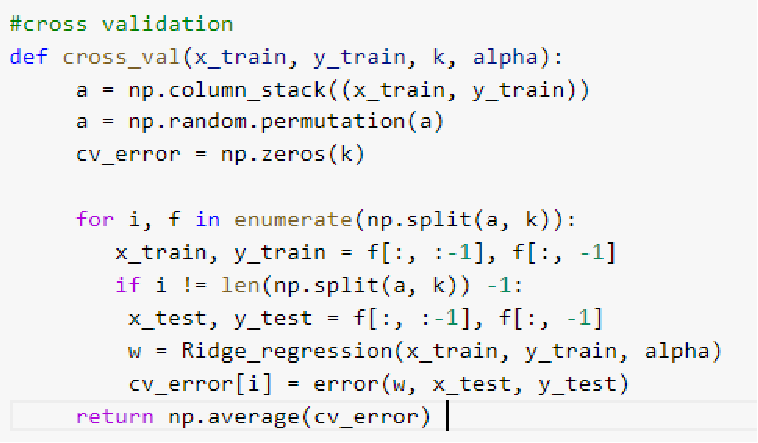
2.2 Implementation and results

After having implemented ridge regression, cross validation and PCA from scratch as follows:



```
#Define ridge regression
def Ridge_regression(X,Y,alpha):
  m = X.shape[0]
  Trans_X=np.transpose(X)
  X_Trans_X=np.dot(Trans_X, X)
  Ident_M=np.identity(X.shape[1])
  ridge=np.linalg.inv(X_Trans_X+(alpha*Ident_M*m))
  t=np.dot(ridge,np.dot(Trans_X,Y))
  return t
#Pred ridge regression
def Ridge_pred(X,w):
  pred =(np.dot(X,w))
  return pred
```

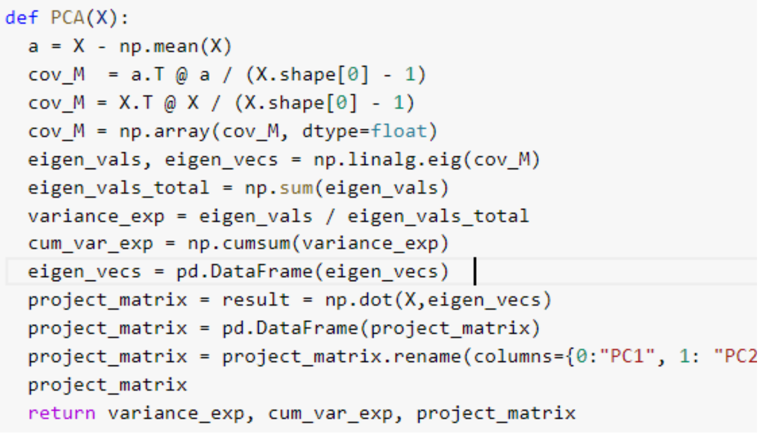
Figure 2.7: Ridge Regression



```
#cross validation
def cross_val(x_train, y_train, k, alpha):
     a = np.column_stack((x_train, y_train))
     a = np.random.permutation(a)
     cv_error = np.zeros(k)

     for i, f in enumerate(np.split(a, k)):
        x_train, y_train = f[:, :-1], f[:, -1]
        if i != len(np.split(a, k)) -1:
         x_test, y_test = f[:, :-1], f[:, -1]
         w = Ridge_regression(x_train, y_train, alpha)
         cv_error[i] = error(w, x_test, y_test)
     return np.average(cv_error)
```

Figure 2.8: Cross Validation



```
def PCA(X):
  a = X - np.mean(X)
  cov_M  = a.T @ a / (X.shape[0] - 1)
  cov_M = X.T @ X / (X.shape[0] - 1)
  cov_M = np.array(cov_M, dtype=float)
  eigen_vals, eigen_vecs = np.linalg.eig(cov_M)
  eigen_vals_total = np.sum(eigen_vals)
  variance_exp = eigen_vals / eigen_vals_total
  cum_var_exp = np.cumsum(variance_exp)
  eigen_vecs = pd.DataFrame(eigen_vecs)
  project_matrix = result = np.dot(X,eigen_vecs)
  project_matrix = pd.DataFrame(project_matrix)
  project_matrix = project_matrix.rename(columns={0:"PC1", 1: "PC2
  project_matrix
  return variance_exp, cum_var_exp, project_matrix
```

Figure 2.9: PCA

Standardization has been applied and 5-fold cross-validation has been implemented to select the best parameter α of ridge regression which reflects the model that probably generalizes better. For each parameter value α, it has been computed the average error over all 5 folds. As shown in results of Fig.2.10, the value of α that minimizes the cross-validation error curve is 0.01 (with CV 0.41).

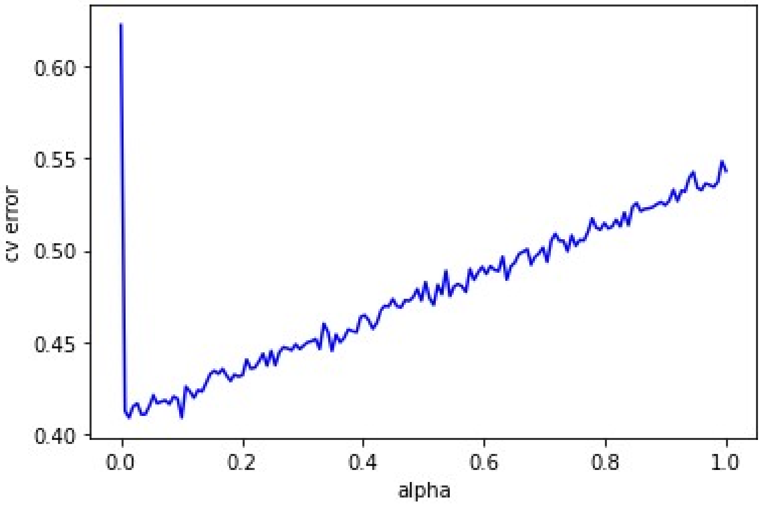


Figure 2.10: Ridge regression results

By limiting the range of α taken, it is possible to notice that the its value minimizing the CV error is 0.011.

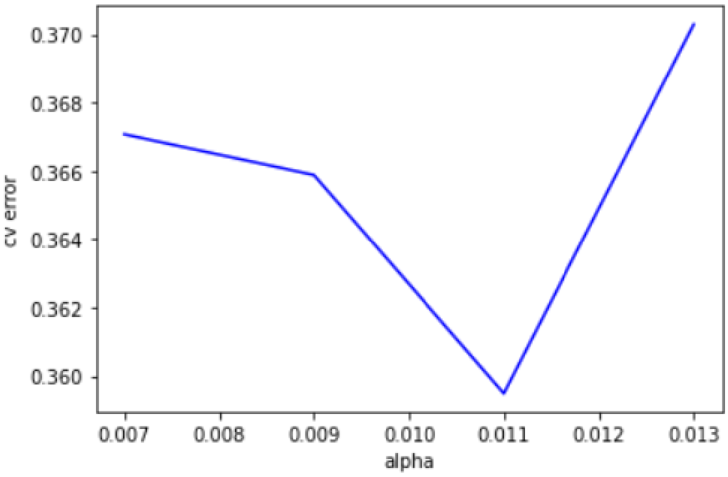
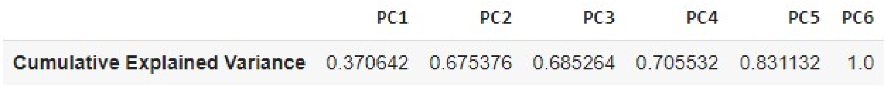


Figure 2.11: Ridge regression results

Alternatively, it is possible to implement PCR and check if it gets better results.
PCA has been applied to feature set, excluding the categorical variable, after having previously scaled them. As shown in Fig. 2.12 and Fig.2.13, the proportion of variance explained by the first 2 components is 0.68, meaning that these PCs capture just 68%

of the variance, explaining just a part of the data. First two PCs have been selected to run ridge regression.



	PC1	PC2	PC3	PC4	PC5	PC6
Cumulative Explained Variance	0.370642	0.675376	0.685264	0.705532	0.831132	1.0

Figure 2.12: PCs and cumulative variance explained

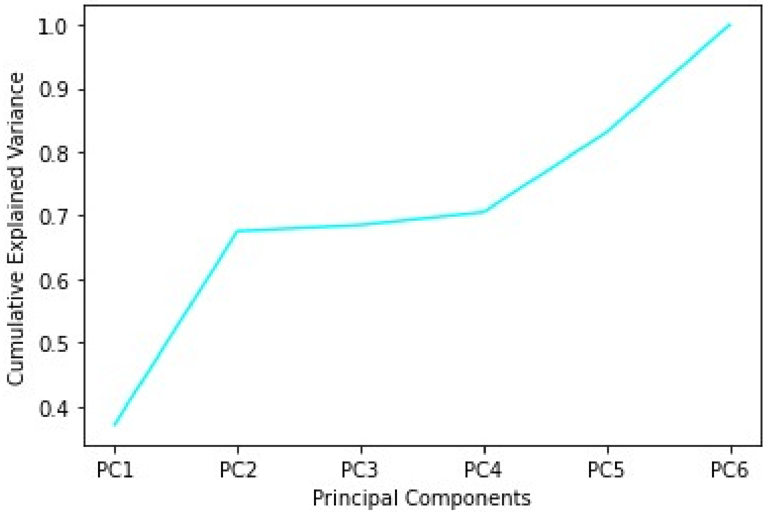


Figure 2.13: Cumulative explained variance

Ridge regression applied to the new modified dataset presents a minimum CV error of 0,98 for α 0.45. Errors increase a lot after applying PCA, almost doubling up.

Conclusions

In this report, it has been implemented ridge regression to predict the median housing price, given various metrics in the Californian Housing Price dataset. Since the data presented multicollinearity among some predictor variables, ridge regression has been considered as a good solution to alleviate the problem. The parameter α of ridge regression has been selected through 5-fold cross-validation. After having implemented both ridge regression on the standardized dataset and on the modified one by PCA, it seems that PCR worsen the overall error. On the other hand, ridge regression seems to be the best model reaching a minimum CV for the regularization parameter $\alpha = 0.01$, presenting the less dispersion.

I declare that this material, which I now submit for assessment, is entirely my own work and has not been taken from the work of others, save and to the extent that such work has been cited and acknowledged within the text of my work. I understand that plagiarism, collusion, and copying are grave and serious offences in the university and accept the penalties that would be imposed should I engage in plagiarism, collusion or copying. This assignment, or any part of it, has not been previously submitted by me or any other person for assessment on this or any other course of study.