Housing Prices

Anna Olena Zhab'yak, Michele Maione

**Summary** –

This paper is trying to answer the question, what is the best model to predict the median house price given the house characteristics aka features.

The classic OLS regression is not enough sometimes to get a good prediction model, for example, if features are highly correlated; it has the desired property of being unbiased, however, it can have a huge variance. In order to pull down the variance, we add some bias with regularization techniques. Using shrinkage Shrinkage is where data values are shrunk towards a central point, like the [mean](https://www.statisticshowto.com/mean/)[[1]](#footnote-1)

In this paper we implement the Ridge regression and the Lasso regression on a dataset of housing prices in order to predict the median house price given the house characteristics. Then PCA is implemented in order to improve the risk estimator.

# Introduction and description of the problem

This paper starts from a real problem of housing prices in the United States, where it is estimated that household wealth is nearly 50% invested in real estate.

63,5% owner-occupied housing rate in July 2019 [[2]](#footnote-2)

The task of predicting the value of a house becomes crucial to the economy as the constructed house price model can influence the growth of the real estate market.

*house price prediction model helps fill up an important information gap and improve the efficiency of the real estate market (Calhoun, 2003)[[3]](#footnote-3).*

An accurate prediction on the house price is important to prospective homeowners, policy makers and other real estate market participants, such as, mortgage lenders and insurers [[4]](#footnote-4)

There are few technical issues with the prediction of the house prices.

The value might be extremely influenced by the value of the neighborhood [[5]](#footnote-5) and so there is a relationship between location and property value, or some exogenous factors can cause an external shock on the overall prices and the prediction becomes wrong.

Moreover working on a large dataset, like the one used in this work, can lead to the so-called multicollinearity of the features which tend to overfit when it comes to implementing certain algorithms to predict the value.

Many works including different techniques have been developed to handle these problems, and they will be introduced later on.

The focus of this paper, however, is on the regularization techniques: the ridge regression, firstly introduced by Hoerl and Kennard (1970), useful tool for improving prediction in regression situations with highly correlated predictors[[6]](#footnote-6);

And Lasso regression

It is a machine learning method that helps to overcome the problem of multicollinearity and therefore to overfitting the data by introducing penalties on the coefficients and the objective function to minimize the overall objective function.

The tuning parameter controls the bias-variance trade-off, so the choice of it is important.

For choosing the regularization parameter in practice, cross-validation (CV) is widely used.

*and the analysis of the risk minimization given different algorithms will be provided*

this two methods can be used on a dataset with features that represent different formats (example here we have money, people, years, geographical information)

# Most important related works

Many works have been developed to predict the median house value, all of them implementing more or less complex models.

For the hedonic model of price prediction, the bundles of characteristics rather than bundles of goods are ranked according to their utility bearing abilities. Attributes (for example, characteristics of a house such as number of bedrooms, number of bathrooms, number of fireplaces, parking facilities, living area and lot size) are implicitly embodied in goods and their observed market prices. The amount or presence of attributes associated with the commodities defines a set of implicit or "hedonic" prices (Rosen, 1974). The marginal implicit values of the attributes are obtained by differentiating the hedonic price function with respect to each attribute (McMillan et al., 1980).

The advantage of the hedonic methods is that they control for the characteristics of properties, thus allowing the analyst to distinguish the impact of changing sample composition from actual property appreciation (Calhoun, 2001)

But

Even though the hedonic price model has been widely recognized, issues such as model specification procedures, multicollinearity, independent variable interactions, heteroscedasticity, non-linearity and outlier data points can seriously hinder the performance of hedonic price model in real estate valuations.[[7]](#footnote-7)

<https://openreview.net/pdf?id=HklRwaEKwB>

Our work is connected to prior works on ridge regression in high-dimensional statistics (Serdobol-skii, 2007

El Karoui (2018)and Dicker (2016) study ridge regression estimators, but focus only on the risk for identity covari-ance. Hastie et al. (2019) study “ridgeless” regression, where the regularization parameter tends tozero.

The influence of the neighbourhood cited before has been studied by Dubin, Robin (2004) and ---

They implemented the kriging algorithm to create an accurate spatial interpolation of house prices in Austin based on the spatial autocorrelation properties of real estate. Implementation of a model to estimate a hedonic regression.

Method of predicting house values is to use data on the characteristics of the area’s housing stock to incorporate for the spatial correlation into the prediction using kriging. [[8]](#footnote-8)

# 3.Notation and relevant definitions (review)

The goal of the regression is to generate a prediction such that the loss function is small for most data points , where is a predictor from the labels set with the regularization term and data domain; the prediction mistakes are a function of the difference.

**Loss function**

To quantify the extent to which the predicted response value for a given observation is close to the true response value for that observation, the most commonly used measure in regression setting is the mean squared error (MSE),

## Quality of fit

The loss function it is used for [parameter estimation](https://en.wikipedia.org/wiki/Parameter_estimation) to map an [event](https://en.wikipedia.org/wiki/Event_(probability_theory)) or values of one or more variables onto a [real number](https://en.wikipedia.org/wiki/Real_number) intuitively representing some "cost" associated with the event. [[9]](#footnote-9)

With loss function we denote the measure of how different the prediction of a hypothesis is from the true outcome.

We use a nonnegative loss function to measure the discrepancy between the predicted label and the true label . In the regression task we define the quadratic loss that is the squared distance between and

When then otherwise If and *c* is large then also tend to be large.

y . {\displaystyle y.} The [risk](https://en.wikipedia.org/wiki/Risk_(statistics)) associated with hypothesis h ( x ) {\displaystyle h(x)} is then defined as the [expectation](https://en.wikipedia.org/wiki/Expected_value) of the loss function:

*The risk function , or the expected loss, becomes the mean squared error of the estimate with respect to the probability distribution*

in regression we refer to the estimation of a probability density as the loss function as the eucleadian distance in a function space , ideed:

And the risk function becomes the squared distances between our target variable and predicted values

The L2 Loss [Mean Square Error (MSE)](https://medium.freecodecamp.org/machine-learning-mean-squared-error-regression-line-c7dde9a26b93) is used as loss function to performe this rigde regression.

## Empirical Risk Minimization

The empirical risk minimization defines a family of learning algorithms used to output some predictor minimizing the training error given a non-negative real-valued loss function :

L ( y ^ , y ) {\displaystyle L({\hat {y}},y)} R ( h ) = E [ L ( h ( x ) , y ) ] = ∫ L ( h ( x ) , y ) d P ( x , y ) . {\displaystyle R(h)=\mathbf {E} [L(h(x),y)]=\int L(h(x),y)\,dP(x,y).} The empirical risk minimization principle states that the learning algorithm should choose a hypothesis h ^ {\displaystyle {\hat {h}}} which minimizes the empirical risk:[[10]](#footnote-10)

**Test Error and training error (**generalization error**)**

est MSE, for a given value x0, can always be decomposed into the sum of three fundamental quantities: the variance of f(x0), the squared bias of f(x0) and the variance of the error terms e. Where, e is the irreducible error, about which we discusses earlier. So, lets see more about bias and variance.

estimate the predictive power of a predictor, we uuse the test error which allows to performe the validation of the algorithm. It is given by

,

The training error is given to fit the model on training set and performe the model selection and it is

Thebiasis an error term that stems from a mismatch between the model class and the under-lying data distribution, and is typically monotonically non-increasing as a function of the complexity of the model.

Where high complexity models could have very low bias, but high variance.

Whereas low complexity models have high bias, but low variance.

model varianceincreases and bias decreasesmonotonicallywith model com-plexity

BIAS VARIANCE TRADE OFF

In any case the prediction will not be perfectly accurate because of the kind of errors that can occur. Total error is given by bias, variance and the irreducible error. Since the latter cannot be improved the variance and bias have to be optimized.

To find

A general rule is that, **as a statistical method tries to match data points more closely or when a more flexible method is used, the bias reduces, but variance increases.**

There is always a trade-oﬀ between these values because it is easy to obtain a method with extremely low bias but high variance (for instance, by drawing a curve that passes through every single training observation) or a method with very low variance but high bias (by ﬁtting a horizontal line to the data). The challenge lies in ﬁnding a method for which both the variance and the squared bias are low.

The bias is the difference between the true population parameter and the expected estimator:

It measures the accuracy of the estimates.



Graph: the optimal choose for trade-off[[11]](#footnote-11)

In Regularization regression we use the hyper parameter alpha to add a bias , prevent the model from real, it will have better accurancy on the testing set, it means it will not overfit in the training set.

Hight bias if the function is non linear or can be nearly linear then my error will be small

If we change exaples, the fit can change significally, it is instable and it leads to variance error. High bias and high variance.

## Regressions – Linear, Ridge, Lasso

Ridge and Lasso regression are part of the family of ERM algorithms. They modify the standard linear regression by introducing a positive constant as a regularization parameter. Indeed the objective function to minimize under these solutions is

**RSS + penalty**

and the penalty is different for the two methods.

Starting from the classical linear model we have

Let be the data domain and

. The linear predictor is a linear function of the data points

, where and where

The Regression with square loss, we want the linear predictor to be as good as possible, for istance, we can look at the risk that would be the base optimal

The empirical risk minimization to is

Since we can rewrite these terms in vector notation, we have

For the vector of predictions and the vector of real labels and for .

We can rewrite everything in matrix notation, let be the design matrix with features and observations we get from data points, that are rows of , and therefore the vector becomes

and applying the ERM we derive

The solution to the ERM is the minimization of this convex function applying the Euclidian distance.

To solve the problem in linear regression we can use closed form solution, or the gradient descend.

If is a non-singular matrix[[12]](#footnote-13), and the conditions of the general position holds, the solution of the ERM is

In some cases, a linear regression performs good in finding a fit on the training data, but it gives a non-accurate estimate on different data. The reason why it occurs it is because of multicollinearity of the prediction vectors (as known as non- orthogonality)[[13]](#footnote-14)

More in general with large or small, the risk that the model can overfit[[14]](#footnote-15) the data is high. The OLS estimator therefore is unbised but have a huge variance and it is not stable.

To overcome this problem, Ridge and Lasso regression help to prevent over-fitting which results from simple linear regression.

We introduce a regularized parameter which adds some bias[[15]](#footnote-16) whereas pushing the variance down.

This occurs to find the best trade-off between bias and variance to get to that sweet spot for having good predictive performance.

The two methods work similarly but lead to different results, this happens because of the divergent formulas.

**Ridge solution**

Ridge regression uses the penalty multiplied by the square of the magnitude of the coefficients, also known as L2 regularization.

The ERM functional of Ridge regression is

,

For , so the solution leads the linear regression, for the coefficient tend to a zero vector and the line becomes flatter, shrinking the linear regression solution towards to zero.

To optimize the objective function, we take the gradient as before and solve for to find a suitable value

The new estimated parameter becomes

This is the so called closed-form solution and is the one measuring the stability of the procedure.

**Lasso solution**

Least Absolute Shrinkage and Selection Operator, or simply Lasso, is slightly different from the previous because the penalty is multiplied by the [absolute value](https://www.statisticshowto.com/integer/#abs) of the magnitude of coefficients, also known as L1 regularization

,

For

The Lasso procedure encourages simple, sparse models[[16]](#footnote-18), indeed some coefficients can become zero and be eliminated from the model, this way performing a feature selection.

The shrinkage amount is given by the value of [tuning parameter](https://www.statisticshowto.com/tuning-parameter/) . If increase, we have some parameters go straightway to zero.

λ (lambda) provides a trade-off between balancing RSS and magnitude of coefficients.

The optimization of a non-differentiable function as Lasso solution is done by a proximal gradient descend approach.

The first step is to take the gradient descend for current vector and form a new vector :

Where is the step size and is the moment we are considering.

Then solve the proximal regularize problem for as follows:

This is a scaler minimization problem indeed we can rewrite it as :

Since we have an absolute value for we consider two cases:

**Case 1**

Differentiate with respect to and solve:

Therefore, since we have the non-negativity constraint over

If ,

Otherwise

**Case 2**

Differentiate with respect to and solve for it:

If

Otherwise

The three solutions are also known as the “soft threshold” operation,

There is some shrinkage going on in this three cases, w is the shank version of z

where the value taken by w is the shrinkage of z

The update rule of the algorithm is therefore

DA COMPLETARE

The common point of these two methods is that adding the regularization parameter to the cost function the algorithm is forced to pick the lowest weights, indeed the goal is to ensure a small coefficient through this regularization parameter.

The main difference is that many coefficients are exactly zeroed under lasso, which is never the case in ridge regression where there is not any elimination of coefficients.

**Cross-validation**

Cross-validation (CV) is one of the techniques used to test the effectiveness of a machine learning models, it is also a re-sampling procedure used to evaluate a model if we have a limited data[[17]](#footnote-19).

In case of limited data, this approach can present a high bias, since there is a substantial loss of data in the training set, but if data is huge and our test sample and train sample has the same distribution then this approach is acceptable.

**K-fold**

K-fold cross-validation is a non-parametric method for evaluating the accuracy of a predictive rule.[[18]](#footnote-20)

it ensures that every observation from the original dataset has the chance of appearing in training and test set

for model selection

Hyperparameters whose value must be determined before the training phase can start

The computed error on the testing part of each fold is

is the dataset divided in subsets , where is the testing part.

We want to estimate the for a fixed hyperparameter.

And the CV final error is the avarage of the errors

given the choice of two predictors, it repeatedly picks the most accurate of the two. Fix a training\test set .

So for tuning the hyperparameter

we look on

where

Estimate risk on all S

This is the algorithm[[19]](#footnote-21)

* for p in 1:P:
* for k in 1:K:
* keep fold k as hold-out data
* use the remaining folds and *λ* = *λp* to estimate β^ridge
* predict hold-out data: ytest,k=Xtest,kβ^ridge
* compute a sum of squared residuals: *S****S****Rk* = ||*y* − *yt****e****st*,*k*||2
* end for k
* average SSR over the folds: SSRp=1K∑k=1KSSRk
* end for p
* choose optimal value: *λo****p****t* = *argminpSSRp*

Moreover the choose of k

he k value must be chosen carefully for your data sample.

A poorly chosen value for k may result in a mis-representative idea of the skill of the model, such as a score with a high variance (that may change a lot based on the data used to fit the model), or a high bias, (such as an overestimate of the skill of the model).

Always remember, a lower value of k is more biased, and hence undesirable. On the other hand, a higher value of K is less biased, but can suffer from large variability. It is important to know that a smaller value of k always takes us towards validation set approach, whereas a higher value of k leads to LOOCV approach.

HOW TO READ BIAS VARIANCE TRADE OFF

After k-fold cross validation, we’ll get *k* different model estimation errors (e1, e2 …..ek). In an ideal scenario, these error values should sum up to zero. To return the model’s bias, we take the average of all the errors. Lower the average value, better the model.

Similarly for calculating the model variance, we take standard deviation of all the errors. A low value of standard deviation suggests our model does not vary a lot with different subsets of training data.

We should focus on achieving a balance between bias and variance. This can be done by reducing the variance and controlling bias to an extent. It’ll result in a better predictive model. This trade-off usually leads to building less complex predictive models as well.

**Principal component analysis**

Principal Component Analysis is a mathematical technique, unsupervised used for dimensionality reduction. Its goal is to reduce the number of features through a combinations of the original data variables. In this way keeping most of the original information.

let's identify the principal components and interpret their relationship to the original variables. The linear coefficients for the PCs (sometimes called the "loadings") are shown in the columns of the Eigenvectors table.

unsupervised algorithms, principal component analysis (PCA)

The features are selected on the basis of variance that they cause in the output.

The feature that causes highest variance is the first principal component.

The feature that is responsible for second highest variance is considered the second principal component, and so on.

It is important to mention that principal components do not have any correlation with each other.

It can be seen that first principal component is responsible for 72.22% variance

A general rule of thumb is to take number of principal of principal components that contribute to significant variance and ignore those with diminishing variance returns. A good way is to plot the variance against principal components and ignore the principal components with diminishing values as shown in the following graph:[[20]](#footnote-22)

VIDEO

We increase the number of features in a intelligent way, because pca keep the most informative.

This way bias is increased but we can have a more stable prediction, it increases the bias but reduces the variance.

One of the points is a linear (ridge)predictor that we have learned.

But basically to inmprove the stability we can use ridge regression as well.

Therefore, ridge regression is a soft PCA regression in fact. They both intend to solve the multi-collinearity in order to improve the model fittness.

**In the PCA analysis negative values of loadings of variable in the components of the PCA means the existence of an inverse correlation between the factor PCA and the variables.**

# 4. Proof of a technical result

The dataset presents features that cannot be compared in a linear Euclidian space; therefore geometry is not working properly on this row data. Indeed, in order to learn the algorithm, we need to encode the features and raise them to a homogeneous level, so we can compare them (ex. Longitude and households are real numbers however they represent different meanings)

Before performing the analysis and regression, pre-procession of data is necessary.

The dataset contains 20640 observations and 10 features for each house including the median house value which is the target value that we are trying to predict. Firstly we create the two constants of labels and data domain:

= median\_house\_value

= data frame

The missing values are handled to avoid errors in the execution of the code.There is a categorical feature which represents the distance from the ocean, we assign a numerical value to each category of labels. The dataset is randomized and outliers are dropped.

It is important to standardize the features by removing the mean and scaling to unit variance.

The L1 (Lasso) and L2 (Ridge) regularizers of linear models assume that all features are centered around 0 and have variance in the same order.

**Model Tuning**

CORRELATION

Finally, looking at Pearson's correlation matrix a features reduction is performed by taking away the most correlated feature with a treshold of |0.75|

Choosing the set of parameters alpha

In order to obtain a reasonable amount of information to determine a certain where is the function that minimize the training error, we use a logarithmic range, .

Training size m is bigger then the in order to avoid underfitting.

**Optimization algorithm**

**Cross- validate risk estimate**

Lasso

Cross-validated risk estimate: 0.022512047975181457

Ridge

Cross-validated risk estimate: 0.02580007952062598

[the plot]

as we see the generalized variance is smaller then in training set as this is the k-fold technque that allows to have this results.

Large a ---------------small a

Hight bias low bias

Low varance high variance

Ex w hat = 0 rss

We perform an analysis of different tunning parameters alpha and the relative mean squared error, comparing the Lasso and Ridge (Cholensky)



The best aplha value is 0.2 more or less for ridge regression, then the error increases (>0.022) for values alpha > 0.2

In lasso regression this increase is smoother, and mse is more stable for alpha in between 0.1 and 1.

This is what we could expect if there are feature highly correlated and that are not crucial for the regression.

On the other hand R2 is speculat to the mse and tells u show well the model fits the data



The coefficient magnitude is shown in this graph



Median income has a huge power in the prediction, it means that it drives the values.

# 5. Some critical considerations

Outliers problem

Even if ridge regressioni s modelled in a way to deal with multicollinearity and outliers, it still being sensitive to outliers. In our work there were some outliers that were pushing the fit in a wrong direction.

<https://iopscience.iop.org/article/10.1088/1742-6596/890/1/012150>

<https://iopscience.iop.org/article/10.1088/1742-6596/890/1/012150/pdf>

As the model complexity increases, the models tends to fit even smaller deviations in the training data set. Though this leads to overfitting, lets keep this issue aside for some time and come to our main objective, i.e. the impact on the magnitude of coefficients. This can be analysed by looking at the data frame created above.

It is clearly evident that the **size of coefficients increase exponentially with increase in model complexity**. I hope this gives some intuition into why putting a constraint on the magnitude of coefficients can be a good idea to reduce model complexity.

What does a large coefficient signify? It means that we’re putting a lot of emphasis on that feature, i.e. the particular feature is a good predictor for the outcome. When it becomes too large, the algorithm starts modelling intricate relations to estimate the output and ends up overfitting to the particular training data.

Lasso can set some coefficients to zero, thus performing variable selection, while ridge regression cannot.

# 6. Copyright

We declare that this material, which We now submit for assessment, is entirely our 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 our work. We 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 us or any other person for assessment on this or any other course of study.

This way Lasso performs better in terms of reducing the variance in models with a lot of features (ritondanti)

In contrast Ridge regression performs better in models where a lot of features are important

# Bibliographical references

* Ramsay J. & Huber M. (1987) - The Techniques of Modern Structural Geology. Volume 2, Folds and Fractures. Academic Press, London, 391 pp.
* Ramsay J. & Huber M. (1987) - The Techniques of Modern Structural Geology. Volume 2, Folds and Fractures. Academic Press, London, 391 pp.

Theory:

<https://arxiv.org/pdf/1509.09169;Lecture>

practice:

<https://www.kirenz.com/post/2019-08-12-python-lasso-regression-auto/>

1. <https://www.statisticshowto.com/lasso-regression/> [↑](#footnote-ref-1)
2. United States Census Bureau

   <https://www.census.gov/quickfacts/fact/table/US/HSG445218> [↑](#footnote-ref-2)
3. Calhoun C. A., 2003, “Property Valuation Models and House Price Indexes for The Provinces ofThailand: 1992 –2000”, Housing Finance International, 17(3): 31 –41. [↑](#footnote-ref-3)
4. Visit Limsombunchai,

   <file:///C:/Users/annao/AppData/Local/Temp/2004-9-house%20price%20prediction.pdf> [↑](#footnote-ref-4)
5. Frew J. and B. Wilson, 2000, “Estimation The Connection Between Location and Property Value”, Essay in Honor of James A.Graaskamp, Boston, MA: Kluwer Academic Publishers [↑](#footnote-ref-5)
6. Riccardo A. Maronna [↑](#footnote-ref-6)
7. Visit Limsombunchai

   <file:///C:/Users/annao/AppData/Local/Temp/2004-9-house%20price%20prediction.pdf> [↑](#footnote-ref-7)
8. Dubin, Robin. (1998). Predicting House Prices Using Multiple Listings Data. The Journal of Real Estate Finance and Economics. 17. 35-59. 10.1023/A:1007751112669. [↑](#footnote-ref-8)
9. <https://en.wikipedia.org/wiki/Loss_function> [↑](#footnote-ref-9)
10. V. Vapnik (1992). [<http://papers.nips.cc/paper/506-principles-of-risk-minimization-for-learning-theory.pdf> *Principles of Risk Minimization* for Learning Theory.*]* [↑](#footnote-ref-10)
11. Source : researchgate.net [↑](#footnote-ref-11)
12. This happens if the data points span [↑](#footnote-ref-13)
13. https://www.math.arizona.edu/~hzhang/math574m/Read/RidgeRegressionBiasedEstimationForNonorthogonalProblems.pdf [↑](#footnote-ref-14)
14. Overfitting: the algorithm performs very good on training data but cannot be generalized to a new bunch of data. [↑](#footnote-ref-15)
15. Bias is how well the fit correspond to the true value [↑](#footnote-ref-16)
16. <https://www.statisticshowto.com/lasso-regression/> [↑](#footnote-ref-18)
17. <https://towardsdatascience.com/why-and-how-to-cross-validate-a-model-d6424b45261f> [↑](#footnote-ref-19)
18. <https://arxiv.org/pdf/1909.11696.pdf> [↑](#footnote-ref-20)
19. <https://www.datacamp.com/community/tutorials/tutorial-ridge-lasso-elastic-net> [↑](#footnote-ref-21)
20. <https://stackabuse.com/implementing-pca-in-python-with-scikit-learn/> [↑](#footnote-ref-22)