Elastic Net Regression

In the end, it's all a question of balance.
- Rohinton Mistry

Outline

Learning Objectives

In this video, you will learn to:

- Understand the cost function of Elastic Net Regression, and the mixing parameter, α.
- Learn to train and evaluate an Elastic Net Regression model in R.
- Compare the regression models that have been introduced so far.

Why do we need Elastic Net Regression?

	Ridge Regression	LASSO Regression
Advantage	Ridge Regression models tend to have slightly higher accuracy.	1. LASSO Regression can perform variable selection. 2. LASSO Regression models have better interpretability.
Disadvantage	1. Ridge Regression models tend to retain all the predictors, and hence, cannot perform variable selection. 2. Ridge Regression models have relatively poorer interpretability.	By variable selection, LASSO Regression may cause some loss of information, resulting in relatively lower accuracy.

- Ridge Regression models work well if most predictors contribute to the dependent variable.
- LASSO Regression models work well if only a few predictors actually influence the dependent variable.
- Elastic Net Regression is a combination of Ridge and LASSO Regression.

Introduction to Elastic Net Regression

Cost Function for Elastic Net Regression

Elastic Net Regression Model

$$Y = \beta_0 + \beta_1 X_1 + \beta_1 X_2 + \cdots + \beta_n X_n$$

- Just like Ridge and LASSO Regression, Elastic Net Regression uses the same type of model as MLR.
- The coefficients of the Elastic Net Regression model are chosen as the ones that minimise the following cost function:

$$\text{Cost Function} = \sum_i \text{Residual}_i^2 + \boldsymbol{\lambda} \sum_{j=1}^{n} \left(\, (1 - \boldsymbol{\alpha})\beta_j^2 + \boldsymbol{\alpha}|\beta_j| \, \right) \qquad \text{where } \boldsymbol{\lambda} \geq 0, \text{and}, 0 \leq \boldsymbol{\alpha} \leq 1.$$

Mixing Parameter: α

- We will refer to α as the ***Mixing Parameter***.
- When $\alpha = 0$, the Elastic Net Regression model will produce the same coefficients as the Ridge Regression model.
- When $\alpha = 1$, the Elastic Net Regression model will produce the same coefficients as the LASSO Regression model.
- When α is strictly between 0 and 1, the Elastic Net Regression model is a mixture of both Ridge and LASSO Regression.

Recap on the glmnet() and cv.glmnet() Functions

```
cv.glmnet(x, y, alpha = M, type.measure = "mse")
glmnet(x, y, alpha = M, lambda = K)
```

The inputs include:

- x is a data matrix of predictor variables, and y is the dependent variable.
- Alpha, with the input M, is the mixing parameter, that determines the type of the Regression model.
- If M is strictly between 0 and 1, the model is trained for Elastic Net Regression.
- "type.measure" indicates the type of error metric on which the models are compared. It can be "MSE", "MAE" or some other metrics.
- Lambda is the regularisation parameter.
- We will use the Cross Validation method to select the optimal α and λ values, that result in the lowest Cross Validation error rate.

Assumptions of Elastic Net Regression Models

Assumptions of Elastic Net Regression Models

1. ***Independence***: Each observation is independent from the others.
2. ***Linearity***: The relationship, between the predictors Xs and the dependent variable Y, is linear.
3. ***Constant Variance***: The residuals are evenly scattered around the center line of zero.

Case Study: Predicting Housing Price

Mr. Tan's Focus Question

What is the expected selling price of houses from one neighbourhood, given the conditions and relevant factors of the area?

Source: `https://www.freepik.com/`

Analyse: Model Building ($\alpha = 0.5$)

- Let us first try $\alpha = 0.5$, and use the cv.glmnet() function to determine the optimal λ, for the Elastic Net Regression model with $\alpha = 0.5$.

```
set.seed(123)
cv_ElaNet0.5 <- cv.glmnet(train.x,train.y, alpha = 0.5, type.measure
    = "mse")
cv_ElaNet0.5
```

```
Call:  cv.glmnet(x = train.x, y = train.y, type.measure = "mse", alpha = 0.5)
Measure: Mean-Squared Error
    Lambda Index Measure     SE Nonzero
min 0.0372    42  0.4235 0.1155       3
1se 0.4180    16  0.5377 0.1224       3
```

- The lambda.min, 0.0372, is the λ value at which the smallest MSE is achieved, among all the Elastic Net models with $\alpha = 0.5$.
- When the λ value equals lambda.min, there are 3 predictors whose coefficients are non-zero.

Analyse: Model Building ($\alpha = 0.5$)

- Let us use the lambda.min and $\alpha = 0.5$ to build the following Elastic Net Regression model.

```
glm_ElaNet0.5 <- glmnet(train.x, train.y, alpha = 0.5,
                        lambda = cv_ElaNet0.5$lambda.min)
t(coef(glm_ElaNet0.5))
```

```
   (Intercept) Crime_rate Industry Number_of_rooms Access_to_highways   Tax_rate
s0   -3.848531 -0.1595763        .       0.7618286                  . -0.1790547
```

- Let us compare the Elastic Net Regression model with the optimal Ridge and LASSO Regression models, that were built in the previous videos.
- Recall that $\alpha = 0$ for Ridge Regression, and $\alpha = 1$ for LASSO Regression.

Compare the Coefficients of three Models ($\alpha = 0, 0.5, 1$)

	Ridge 0	ElasticNet 0.5	LASSO 1
(Intercept)	-3.60296342	-3.8485307	-3.9440373
Crime_rate	-0.16533979	-0.1595763	-0.1574161
Industry	0.00360547	.	.
Number_of_rooms	0.73478112	0.7618286	0.7718826
Access_to_highways	-0.02070860	.	.
Tax_rate	-0.17184176	-0.1790547	-0.1773956

- The coefficients of the Elastic Net Regression model, with $\alpha = 0.5$, is very close to that of the LASSO Regression model, where $\alpha = 1$.
- Both the Elastic Net Regression model and the LASSO Regression model have performed variable selection, by keeping coefficients of two predictors as zero.

Extract the lambda.min and MSE Values

```
Call:  cv.glmnet(x = train.x, y = train.y, type.measure = "mse", alpha = 0.5)
Measure: Mean-Squared Error
    Lambda Index Measure     SE Nonzero
min 0.0372    42  0.4235 0.1155       3
1se 0.4180    16  0.5377 0.1224       3
```

```
cv_ElaNet0.5$lambda.min
```

```
[1] 0.03721264
```

```
cv_ElaNet0.5$cvm [cv_ElaNet0.5$lambda == cv_ElaNet0.5$lambda.min]
```

```
[1] 0.4234997
```

Compare the MSE Rates between Ridge, Elastic Net, and LASSO Regression Models

	alpha	lambda.min	MSE
Ridge	0	0.084	0.433
Elastic Net	0.5	0.037	0.423
LASSO	1	0.022	0.423

- The Elastic Net Regression model and the LASSO Regression model have lower Cross Validation error rates, and perform better than the Ridge Regression model, on the training dataset.
- Among the options with α values being 0, 0.5 and 1, the optimal pair of α and λ values, are either ($\alpha = 0.5$, $\lambda = 0.037$), or ($\alpha = 1$, $\lambda = 0.022$).

Apply the Cross Validation Method to Select the Optimal Parameters

Apply the Cross Validation Method

- We apply the Cross Validation method to all possible pairs of (α, λ), and choose the optimal pair of parameters, that achieves the minimum Cross Validation error rate among all.
- We will create 11 optimal models for different α values, ranging from 0, 0.1, $\cdots$, to 0.9, and 1.

```
generate_cvmodels <- function (x) {
  set.seed(123)
  return(cv.glmnet(train.x, train.y,
                   type.measure = "mse", alpha = x/10))
}
cv_models <- lapply(0:10, generate_cvmodels)
```

- Note that each α value corresponds to an unique lambda.min value.

Select the Model with the Lowest MSE

- Next, we will compare the 11 models to select the one with the lowest Cross Validation MSE.

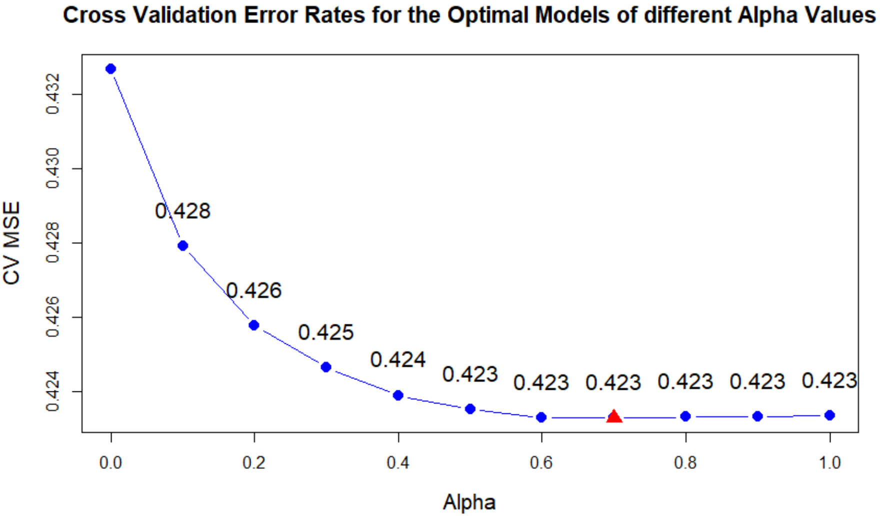
```
cv_error <- unlist(lapply(cv_models, function(x) x$cvm[x$lambda ==
    x$lambda.min] ))
(which(cv_error == min(cv_error)) -1)/10
```

```
[1] 0.7
```

- The optimal α is 0.7. We can visualise it in the following plot of MSE rates across different α values.

```
alpha_seq <- c(0:10)/10
plot(alpha_seq, cv_error, main = "Cross Validation Error Rates for
    the Optimal Models of different alpha Values",
       xlab = "Alpha", ylab = "CV MSE", cex = 1.4, cex.lab = 1.3,
       cex.main = 1.4, pch = 19, col = "blue", type = "b")
text(alpha_seq, cv_error + 0.001, labels=round(cv_error, digits = 3),
     cex = 1.3)
```

Cross Validation Error Rates vs. Alpha



Get the Optimal Pair of Parameters

- Let us use the following code chunk to fetch the optimal pair of parameters, that achieves the lowest MSE among all.

```
get_best_model <- function (models, errors) {  #models is a list of
    models, and errors is a list of errors
  best_n <- which(errors == min(errors))
  return(
    data.frame(
      alpha = (best_n - 1)/10,  # best_n = 1 refers to alpha = 0, and
          best_n = 11 refers to alpha = 1.
      lambda = models[[best_n]]$lambda.min,
      CV_error = errors[best_n]
    )
  )
}
best_parameter <- get_best_model(cv_models, cv_error)
best_parameter
```

Get the Optimal Pair of Parameters

```
  alpha        lambda     CV_error
1   0.7    0.02917202    0.4232769
```

- The optimal pair of parameters is ($\alpha = 0.7$, $\lambda = 0.0292$).
- The corresponding Cross Validation MSE is 0.4233, which is the minimum MSE among all.

Build the Final Elastic Net Regression Model

Analyse: Build the Final Elastic Net Regression Model

```
glm_ElaNet <- glmnet(train.x, train.y, alpha = best_parameter$alpha,
                     lambda = best_parameter$lambda)
t(coef(glm_ElaNet))
```

```
   (Intercept) Crime_rate Industry Number_of_rooms Access_to_highways   Tax_rate
s0   -3.899789 -0.1585537        .       0.7672639                  . -0.1782727
```

- Like LASSO Regression, the final Elastic Net Regression model has performed variable selection, by setting the coefficients of two predictors, "Industry" and "Access to highways", as zero.

Evaluate All the Regression Models

Build the Baseline MLR Model

- We build our baseline MLR model by removing two predictors, "Industry" and "Access to highways".

```
mlr_adj <- lm(Price ~. -Access_to_highways - Industry, data =
    training)
summary(mlr_adj)
```

```
Call:
lm(formula = Price ~ . - Access_to_highways - Industry, data = training)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.18352 -0.32599 -0.03098  0.18866  2.85090 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)     -4.06558    0.68939  -5.897 5.38e-08 ***
Crime_rate      -0.16705    0.07135  -2.341   0.0213 *  
Number_of_rooms  0.78930    0.07070  11.164  < 2e-16 ***
Tax_rate        -0.18882    0.07785  -2.425   0.0171 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6352 on 97 degrees of freedom
Multiple R-squared:  0.6783,	Adjusted R-squared:  0.6684 
F-statistic: 68.19 on 3 and 97 DF,  p-value: < 2.2e-16
```


Analyse: Evaluate All the Regression Models on the Training Dataset

	MSE	MAE	RMSE	MAPE	R^2
MLR Adjusted Train	0.388	0.424	0.623	0.210	0.678
Ridge Train	0.391	0.423	0.625	0.208	0.676
LASSO Train	0.388	0.424	0.623	0.210	0.678
Elastic Net Train	0.389	0.424	0.623	0.209	0.678

From the summary table, we can see:

- Across all the error metrics on the ***training dataset***, the four Regression models perform equally well.
- We get similar conclusions from R^2.

Analyse: Evaluate All the Regression Models on the Test Dataset

	MSE	MAE	RMSE	MAPE
MLR Adjusted Test	0.482	0.444	0.694	0.227
Ridge Test	0.461	0.439	0.679	0.222
LASSO Test	0.472	0.441	0.687	0.224
Elastic Net Test	0.470	0.440	0.686	0.224

From the summary table, we can see:

- The adjusted MLR model works the poorest, as its MSE, 0.482, is the highest among all.
- The three regularised models outperform the adjusted MLR model, on the ***test dataset***.

Regularised Models have Minimised the Overfitting Problem

	Train MSE	Test MSE
MLR Adjusted	0.388	0.482
Ridge	0.391	0.461
LASSO	0.388	0.472
Elastic Net	0.389	0.470

- The MSE of the adjusted MLR model, on the test dataset, is 24.2% higher than that on the training dataset.
- The MSE of the Ridge Regression model, on the test dataset, is only 17.9% higher than that on the training dataset.
- We have similar observations for the LASSO and Elastic Net Regression models, with the percentage differences, 21.6%, and 20.8%, respectively.
- In other words, the Regularised models, the Ridge, LASSO and Elastic Net Regression models have minimised the Overfitting problem, by reducing the differences of the error metrics between the training and the test datasets.

Ridge, LASSO and Elastic Net Regression

- Both LASSO and Elastic Net Regression models have performed the variable selection.
- As a mixture of Ridge and LASSO, the Elastic Net model has combined their strength, namely, it not only improves the model interpretability, but also maintains a good level of accuracy.
- If you prioritise accuracy, Ridge Regression will be suitable.
- If you value the business interpretability, LASSO Regression will be a better choice.
- If you wish to achieve a good balance between accuracy and interpretability, Elastic Net Regression is recommended.

Summary

Summary

We have learnt to:

- Understand how an Elastic Net Regression model works.
- Understand how the mixing parameter, α, impacts the Elastic Net Regression model.
- Can apply the Cross Validation method to select the optimal pair of parameters, (α, λ), and use them to train an Elastic Net Regression model in R.
- Understand the strength and limitations of various Regression models. In particular, understand how the Regularisation methods solve the Multicollinearity and Overfitting problems.
- The Regularisation methods can be applied to not only the Multiple Linear Regression Models, but also some other Regression models, for example, the Logistic Regression, Poisson Regression and Negative Binomial Regression models.

References

- Hastie, Qian, and Tay (2021), *An Introduction to glmnet*
  https://glmnet.stanford.edu/articles/glmnet.html
- Dataset: the Boston Housing Dataset
  https://www.cs.toronto.edu/ delve/data/boston/bostonDetail.html
- Shubham.jain Jain (2017), *A comprehensive beginners guide for Linear, Ridge and Lasso Regression in Python and R*
  https://www.analyticsvidhya.com/blog/2017/06/a-comprehensive-guide-for-linear-ridge-and-lasso-regression/