



George Seif

[Follow](#)

Certified Nerd. Lover of learning. AI / Machine Learning Engineer.

Mar 26 · 6 min read

• 5 Types of Regression and their properties

Linear and Logistic regressions are usually the first modelling algorithms that people learn for Machine Learning and Data Science. Both are great since they're easy to use and interpret. However, their inherent simplicity also comes with a few drawbacks and in many cases they're not really the best choice of regression model. There are in fact several different types of regressions, each with their own strengths and weaknesses.

In this post, we're going to look at 7 of the most common types of regression algorithms and their properties. We'll soon find that many of them are biased to working well in certain types of situations and with certain types of data. In the end, this post will give you a few more tools in your regression tool box and give greater insight into regression models as a whole!

Linear Regression

Regression is a technique used to model and analyze the relationships between variables and often times how they contribute and are related to producing a particular outcome together. A linear regression refers to a regression model that is completely made up of linear variables. Beginning with the simple case, Single Variable Linear Regression is a technique used to model the relationship between a single input independent variable (feature variable) and an output dependent variable using a linear model i.e a line.

The more general case is Multi Variable Linear Regression where a model is created for the relationship between multiple independent input variables (feature variables) and an output dependent variable. The model remains linear in that the output is a linear combination of the input variables. We can model a multi-variable linear regression as the following:

$$Y = a_1 * X_1 + a_2 * X_2 + a_3 * X_3 + \dots + a_n * X_n + b$$

Where a_n are the coefficients, X_n are the variables and b is the bias.

As we can see, this function does not include any non-linearities and so is only suited for modelling linearly separable data. It is quite easy to understand as we are simply weighting the importance of each feature variable X_n using the coefficient weights a_n . We determine these weights a_n and the bias b using a Stochastic Gradient Descent (SGD). Check out the illustration below for a more visual picture!

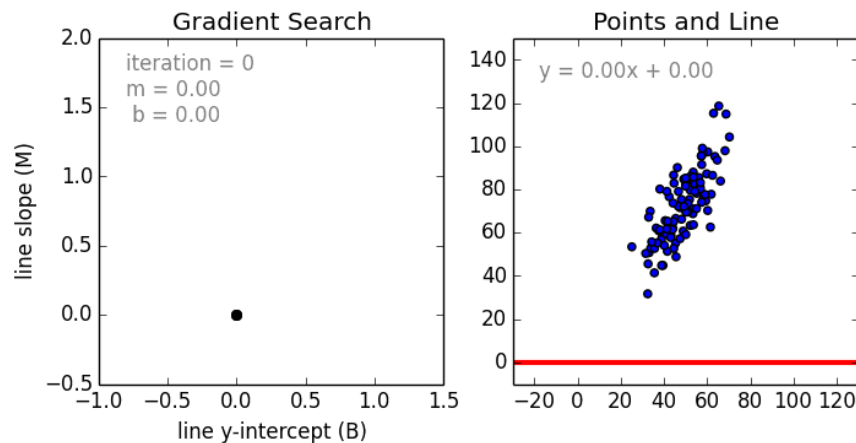


Illustration of how Gradient Descent find the optimal parameters for a Linear Regression

A few key points about Linear Regression:

Fast and easy to model and is particularly useful when the relationship to be modeled is not extremely complex and if you don't have a lot of data.

Very intuitive to understand and interpret.

Linear Regression is very sensitive to outliers.

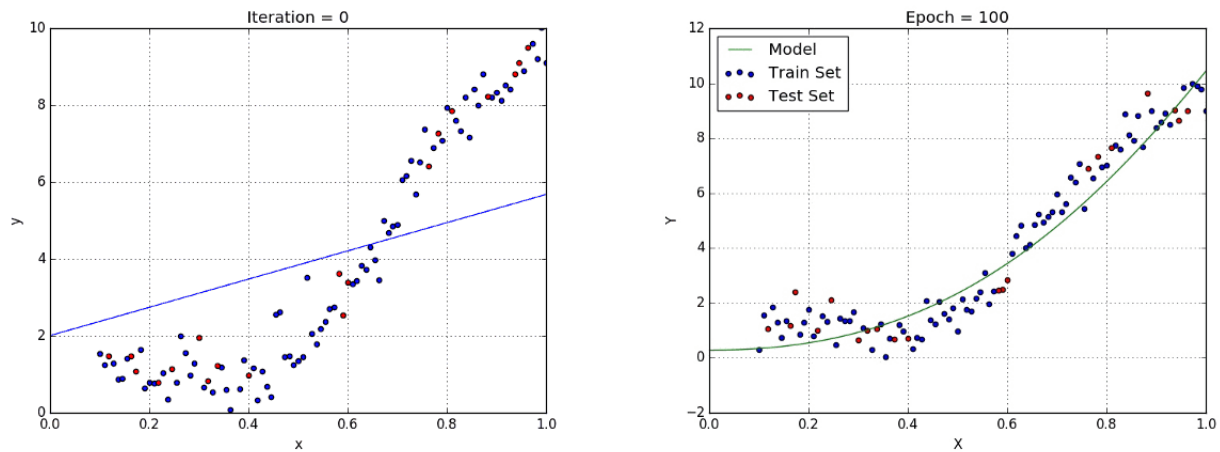
Polynomial Regression

When we want to create a model that is suitable for handling non-linearly separable data, we will need to use a polynomial regression. In this regression technique, the best fit line is not a straight line. It is rather a curve that fits into the data points. For a polynomial regression, the power of some independent variables is more than 1. For example, we can have something like:

$$Y = a_1 * X_1 + (a_2)^2 * X_2 + (a_3)^4 * X_3 \dots a_n * X_n + b$$

We can have some variables have exponents, others without, and also select the exact exponent we want for each variable. However, selecting the exact exponent of each variable naturally requires some

knowledge of how the data relates to the output. See the illustration below for a visual comparison of linear vs polynomial regression.



Linear vs Polynomial Regression with data that is non-linearly separable

- A few key points about Polynomial Regression:
 - Able to model non-linearly separable data; linear regression can't do this. It is much more flexible in general and can model some fairly complex relationships.
 - Full control over the modelling of feature variables (which exponent to set).
 - Requires careful design. Need some knowledge of the data in order to select the best exponents.
 - Prone to over fitting if exponents are poorly selected.

Ridge Regression

A standard linear or polynomial regression will fail in the case where there is high collinearity among the feature variables. Collinearity is the existence of near-linear relationships among the independent variables. The presence of high collinearity can be determined in a few different ways:

- A regression coefficient is not significant even though, theoretically, that variable should be highly correlated with Y.
- When you add or delete an X feature variable, the regression

coefficients change dramatically.

Your X feature variables have high pairwise correlations (check the correlation matrix).

We can first look at the optimization function of a standard linear regression to gain some insight as to how ridge regression can help:

$$\min ||Xw - y||^2$$

Where X represents the feature variables, w represents the weights, and y represents the ground truth. Ridge Regression is a remedial measure taken to alleviate collinearity amongst regression predictor variables in a model. Collinearity is a phenomenon in which one feature variable in a multiple regression model can be linearly predicted from the others with a substantial degree of accuracy. Since the feature variables are so correlated in this way, the final regression model is quite restricted and rigid in its approximation i.e it has high variance.

To alleviate this issue, Ridge Regression adds a small squared bias factor to the variables:

$$\min ||Xw - y||^2 + \lambda ||w||^2$$

Such a squared bias factor pulls the feature variable coefficients away from this rigidity, introducing a small amount of bias into the model but greatly reducing the variance.

A few key points about Ridge Regression:

The assumptions of this regression is same as least squared regression except normality is not to be assumed.

It shrinks the value of coefficients but doesn't reaches zero, which suggests no feature selection feature

Lasso Regression

Lasso Regression is quite similar to Ridge Regression in that both techniques have the same premise. We are again adding a biasing term to the regression optimization function in order to reduce the effect of collinearity and thus the model variance. However, instead of using a squared bias like ridge regression, lasso instead using an absolute value bias:

$$\min ||Xw - y||^2 + z ||w||$$

There are a few differences between the Ridge and Lasso regressions that essentially draw back to the differences in properties of the L2 and L1 regularization:

Built-in feature selection: is frequently mentioned as a useful property of the L1-norm, which the L2-norm does not. This is actually a result of the L1-norm, which tends to produce sparse coefficients. For example, suppose the model has 100 coefficients but only 10 of them have non-zero coefficients, this is effectively saying that “the other 90 predictors are useless in predicting the target values”. L2-norm produces non-sparse coefficients, so does not have this property. Thus one can say that Lasso regression does a form of “parameter selection” since the feature variables that aren’t selected will have a total weight of 0.

Sparsity: refers to that only very few entries in a matrix (or vector) is non-zero. L1-norm has the property of producing many coefficients with zero values or very small values with few large coefficients. This is connected to the previous point where Lasso performs a type of feature selection.

Computational efficiency: L1-norm does not have an analytical solution, but L2-norm does. This allows the L2-norm solutions to be calculated computationally efficiently. However, L1-norm solutions do have the sparsity properties which allows it to be used along with sparse algorithms, which makes the calculation more computationally efficient.

ElasticNet Regression

ElasticNet is a hybrid of Lasso and Ridge Regression techniques. It uses both the L1 and L2 regularization taking on the effects of both techniques:

$$\min ||Xw - y||^2 + z_1 ||w|| + z_2 ||w||^2$$

A practical advantage of trading-off between Lasso and Ridge is that, it allows Elastic-Net to inherit some of Ridge’s stability under rotation.

A few key points about ElasticNet Regression:

It encourages group effect in the case of highly correlated variables, rather than zeroing some of them out like Lasso.

There are no limitations on the number of selected variables.

Conclusion

There you have it! 5 common types of Regressions and their properties. All of these regression regularization methods (Lasso, Ridge and ElasticNet) work well in case of high dimensionality and multicollinearity among the variables in the data set. I hope you enjoyed this post and learned something new and useful. If you did, feel free to give it some claps.

