What is Penalized Regression?

Overview: Penalized regression is a variation of regression that can help with highly dimensional data, correlated predictors, and overfitting.

Motivation:

Let's look at a simple linear regression example relating hours of studying to exam score.

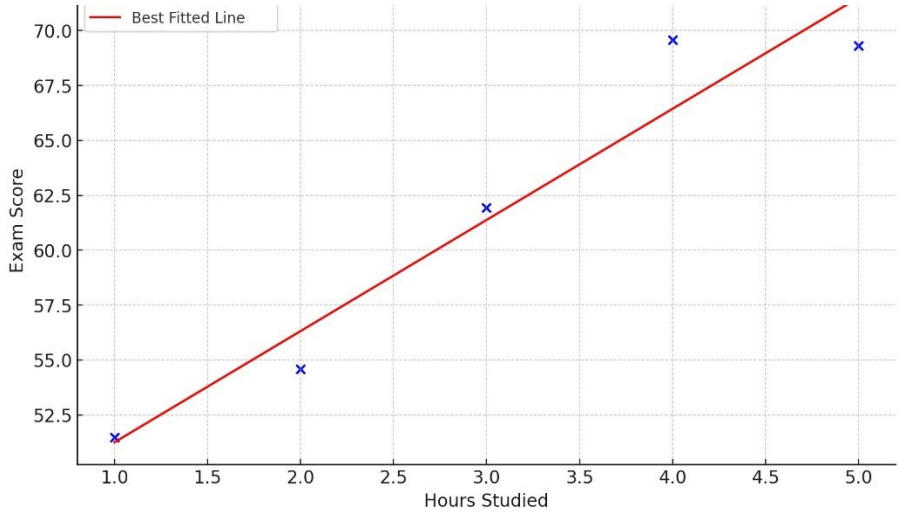

Figure 1: Example data set relating hours studies (x) to exam score (y)

We can see generally that as expected, as the number of hours studied increased, the exam score increased. The red line represents the linear regression model that was estimated based on the data. The model we fit was a simple model:

$$\text{Exam score} = \beta_0 + \beta_1 \text{ x } \textit{hours of studying}$$

How do we obtain the best fitting regression line in general (i.e.: estimate the β_0 and β_1)? We minimize something called a *cost function*.

In linear regression, this cost function is the squared error (i.e.: the squared difference between the predicted y and the actual y), summed across all observations.

Cost function = sum of squared errors = $\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$

To find the best slope and intercept we can compute this cost function across many possible slopes and intercept values to see where the minimum occurs.

Models have a tendency to be **overfit to the data set** in which they were trained. An *overfit model* means that it has low error because it is highly tuned to the nuances of the data set it was trained on but is not actually reflecting the true relationship between x and y, and therefore will fail to generalize to new instances well. **A clear sign that you have an overfit model is getting a very close fit to the original data with low error, but then observing high error (lack of fit) when applying the model to new data sets.**

Overfitting commonly occurs when:

1. Sample sizes are too small. The estimated model tends to fit the patterns observed in the data. When you have a small number of observations, these patterns are unstable and not well-supported, thus are likely not going to be consistently observed in different data sets
2. Model complexity is too high. Complexity means we are including quadratic terms, interactions, or too many unimportant predictors in the model – this high dimensionality means we have lots of ways to closely fit the noise in our data set that may not generalize to other data sets.

3. High levels of correlation between subsets of predictors. If X1 and X2 are highly correlated in our original data set and they are both included in the model, this may not generalize well if such correlation does not hold in other data sets.

Let's look at an example of a model of where overfitting is highly likely. This is a complicated regression model where we have added polynomial terms.

Exam score =

$\beta 0 + \beta 1 \times \text{hours studied} + \beta 2 \times \text{hours studied}^2 + \beta 3 \times \text{hours studied}^3 + \beta 4 \times \text{hours studied}^4$

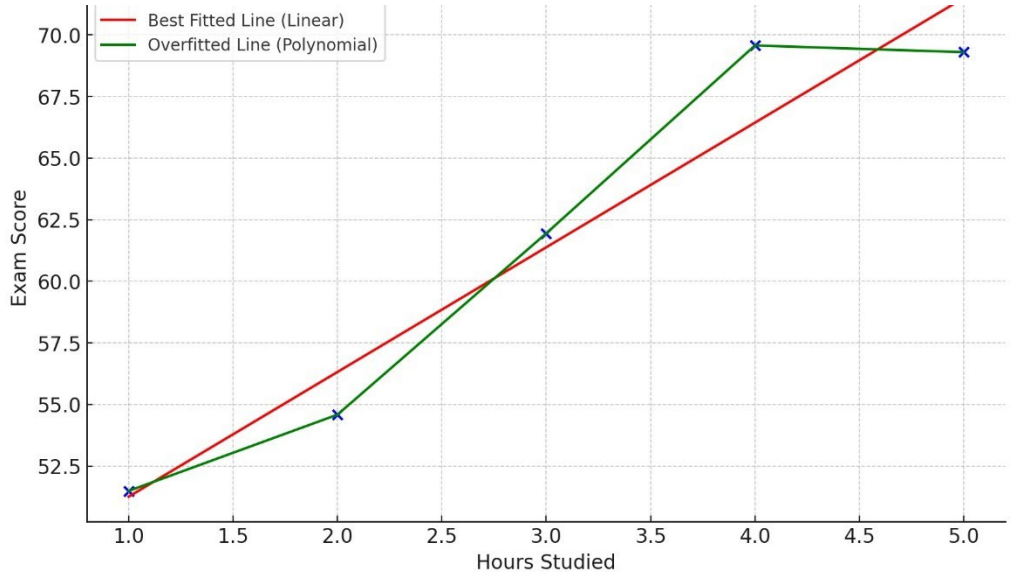

Figure 2: Example of over-fitting (green line). As you can see, the overfit model is very likely capturing too many nuances of the training data set to be generalizable. In this case, the original model containing only study time as a predictor provides a more stable and less overfit model.

This model, while having a cost function value that is 0 (because the predicted function is passing through all observed points), is clearly too complex. The true relationship between hours studied and exam score is probably not nearly as complicated as this.

Penalization

To mitigate overfitting and prevent extreme coefficient estimates that are too closely aligned with the training data set, we can modify the cost function by incorporating **penalization**. This technique adjusts the cost function to **balance model fit with the total coefficient magnitude**, reducing the likelihood of overfitting.

In performing penalized regression, we change the cost function:

$$\textbf{Cost function} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \textbf{penalty}$$

So instead of just trying to minimize the squared errors (and getting as close to the observed data as possible), we also now try to minimize a penalty at the same time. The penalty is going to be a function of the estimated coefficients in the model (the βs).

There are two penalties that are typically used in penalized regression:

LASSO (Least Absolute Shrinkage and Selection Operator; L1 Penalty)

$$\textbf{Cost function} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{i=1}^{k} |\beta_i|$$

Usual cost function for linear regression + penalty

Ridge (L2 Penalty)

$$\textbf{Cost function} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{i=1}^{k} \beta_i^2$$

Although the penalties used for LASSO and Ridge regression are similar, they can yield very different solutions. **LASSO tends to produce sparse models** by driving some coefficients exactly to zero, effectively performing variable selection in addition to coefficient shrinkage. In contrast, **Ridge regression shrinks coefficients towards zero but typically all predictors have a non-zero coefficient**. Therefore, while Ridge regression reduces the magnitude of coefficients, it does not eliminate predictors, whereas LASSO can both shrink coefficients and eliminate uninformative predictors, leading to a simpler and more interpretable model.

Let's take a step back and get a general sense of what is going on with LASSO regression. In **Figure 3** I have graphed the contours of a sum of squared errors cost function. These contours reflect combinations of coefficients for the model that produce the same sum of squared errors. If we were going to fit a standard regression, we would try to minimize this cost function, and thus our optimal solution would be found in the very center of the contours. **But we know from our exam score example that adding in a bunch of complex terms to the model and the minimizing the cost function to find the best estimated model coefficients can produce a model that is overfit to the data at hand**. We thus impose a penalty on this process – we want to minimize cost but also not let the sum of the absolute values of the coefficients get too large (either due to large estimates in a few predictors and/or too many coefficients). Superimposed on these contours is the LASSO penalty. As you can see, it is a diamond-shaped boundary (because we are constraining the sum of absolute values – if we use ridge regression that squares the model coefficients this boundary will be a circle) with a size that depends on

the lambda value we specify. **If lambda (λ) is large, we are imposing a higher penalty on the magnitude of the estimated coefficients and the diamond will be smaller**. And as the diamond gets smaller and smaller we are basically tolerating more squared error loss (because the contour it intersects will be further and further from the center) in order to impose a more stringent constraint on the coefficients.

To find an optimal solution, we would look to see where the sum of squared errors cost intersects with the LASSO penalty.

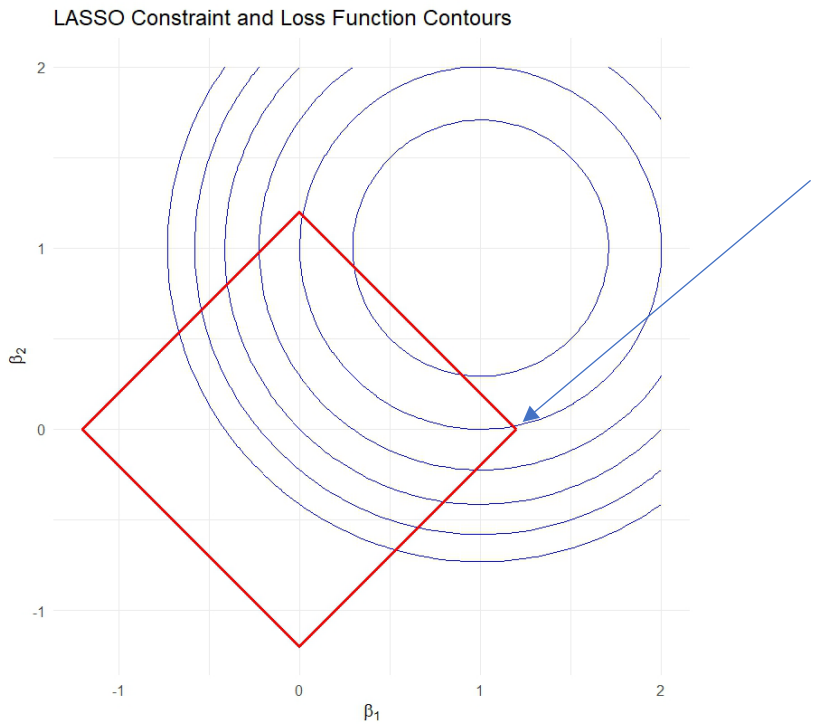

Figure 3: Contours of the SSE cost function values with LASSO penalty

In LASSO, the intersection between the contours and penalty may typically be at the corners of the diamond, meaning that some coefficients can be set exactly to zero. In Ridge regression, the penalty (summing the squares of the coefficients instead of the

absolute values) is **a circle instead of a diamond, which means that the optimal solution does not usually set coefficients exactly equal to zero**.

How does this help with overfitting? When you overfit your data set, the coefficients are typically more extreme, and the model is far more complex than it should be in order to fit all of the nuances in your data. **Using penalized regression allows you to fit a model with minimum error while ensuring that the complexity of the model is reasonable and will be able to generalize to other data sets.**

Where can I read more about this?

There are many **excellent online resources** that will allow you to more fully understand.