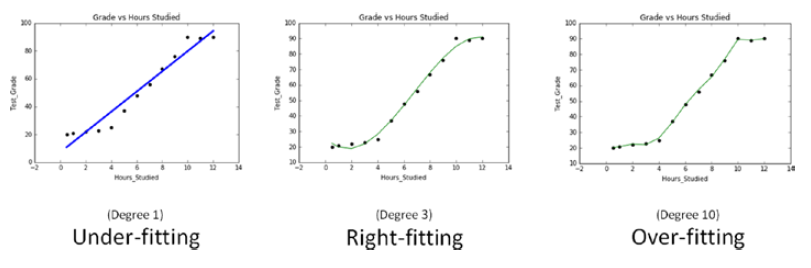
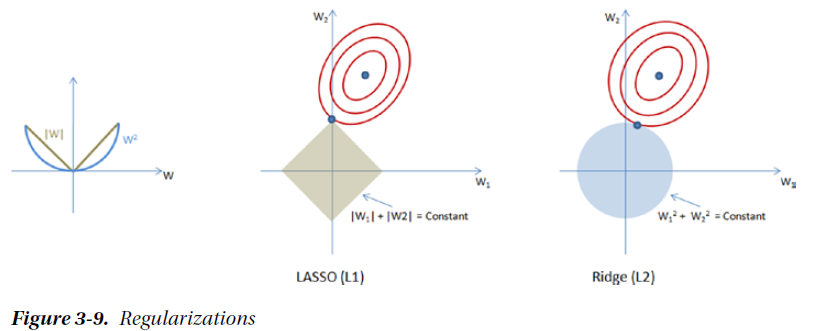
**Over-fitting and Under-fitting:**

Under-fitting occurs when the model does not fit the data well and is unable to capture the underlying trend in it. In this case we can notice a low accuracy in training and test dataset. To the contrary, over-fitting occurs when the model fits the data too well, capturing all the noises. In this case we can notice a high accuracy in the training dataset, whereas the same model will result in a low accuracy on the test dataset. This means the model has fitted the line so well to the train dataset that it failed to generalize it to fit well on an unseen dataset. Figure 3-8 shows how the different fitting would look like on the earlier discussed example use case. The choice of right order polynomial degree is very important to avoid an over-fitting or under-fitting issue in regression. We’ll also discuss in detail about different ways of handling these problems in the next chapter.



**The Bias-Variance Trade-off**: Another way of thinking about the overfitting problem is as a trade-off between bias and variance. Both are measures of what would happen if you were to retrain your model many times on different sets of training data (from the same larger population). For example, the degree 0 model in “Overfitting and Underfitting” on page 142 will make a lot of mistakes for pretty much any training set (drawn from the same population), which means that it has a high bias. However, any two randomly chosen training sets should give pretty similar models (since any two randomly chosen training sets should have pretty similar average values). So we say that it has a low variance. High bias and low variance typically correspond to underfitting. On the other hand, the degree 9 model fit the training set perfectly. It has very low bias but very high variance (since any two training sets would likely give rise to very different models). This corresponds to overfitting. Thinking about model problems this way can help you figure out what do when your model doesn’t work so well. If your model has high bias (which means it performs poorly even on your training data) then one thing to try is adding more features. Going from the degree 0 model in “Overfitting and Underfitting” to the degree 1 model was a big improvement. If your model has high variance, then you can similarly remove features. But another solution is to obtain more data (if you can).

**Regularization:**With an increase in number of variables, and increase in model complexity, the probability of over-fitting also increases. Regularization is a technique to avoid the over-fitting problem. Over-fitting occurs when the model fits the data too well, capturing all the noises. In this case we can notice a high accuracy in the training dataset, whereas the same model will result in a low accuracy on the test dataset. This means the model has fitted the line so well to the train dataset that it failed to generalize it to fit well on the unseen dataset. Statsmodel and the scikit-learn provides Ridge and LASSO (Least Absolute Shrinkage and Selection Operator) regression to handle the over-fitting issue. With an increase in model complexity, the size of coefficients increase exponentially, so the ridge and LASSO regression apply penalty to the magnitude of the coefficient to handle the issue. **LASSO:** This provides a sparse solution, also known as **L1 regularization**. It guides parameter value to be zero, that is, the coefficients of the variables that add minor value to the model will be zero, and it adds a penalty equivalent to absolute value of the magnitude of coefficients. **Ridge Regression**: Also known as Tikhonov **L2 regularization**, it guides parameters to be close to zero, but not zero. You can use this when you have many variables that add minor value to the model accuracy individually; however it improves overall the model accuracy and cannot be excluded from the model. Ridge regression will apply a penalty to reduce the magnitude of the coefficient of all variables that add minor value to the model accuracy, and which adds penalty equivalent to square of the magnitude of coefficients. Alpha is the regularization strength and must be a positive float.  

**Regularization basically adds the penalty as model complexity increases.** **Regularization parameter (lambda) penalizes all the parameters except intercept so that model generalizes the data and won’t overfit.**

