CSE 6363 - Machine Learning

Regularization

Alex Dillhoff

University of Texas at Arlington

What is covered?

1. Overfitting
2. Least-Squares Regularization
3. Evaluation on polynomial model

Overfitting

What happens when the complexity of our model fits the data *too* well?

A highly parameterized model may seem desirable, but can lead to worse performance than a simple one.

Overfitting

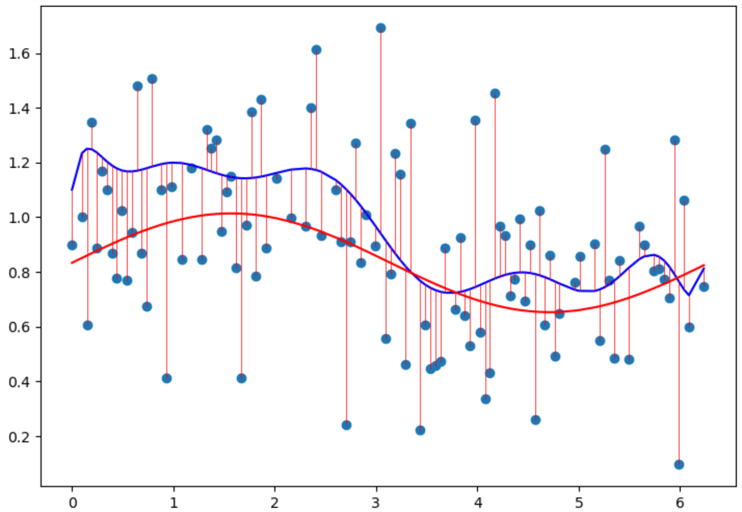

Figure 1: A polynomial of degree 11 (blue) fit to data generated following the red curve.

Overfitting

The model with more parameters is able to fit some the noisy data slightly better.

Does this necessarily mean it will perform better on new samples?

Overfitting

No, it will usually perform worse.

This is referred to as **overfitting.**

Overfitting

Models with more parameters have higher capacity to fit the complexity of a data.

However, much of that complexity may just be simple noise.

This the model will not **generalize** well to unseen data.

Overfitting

Overfitting can be detected in many ways.

When using online learning, tracking the training loss versus the validation loss reveals when the model begins to overfit.

Overfitting

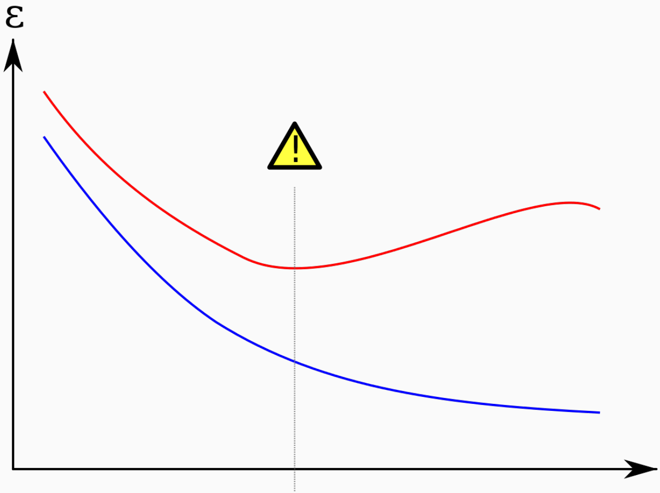

Figure 2: The validation loss (red) diverges as the training loss (blue) continues to decrease. Source: Wikipedia

Overfitting

If using a different solver, you can inspect the weights of the model.

Left unchecked, weights of a highly parameterized model usually take on large values as the loss is minimized.

Overfitting

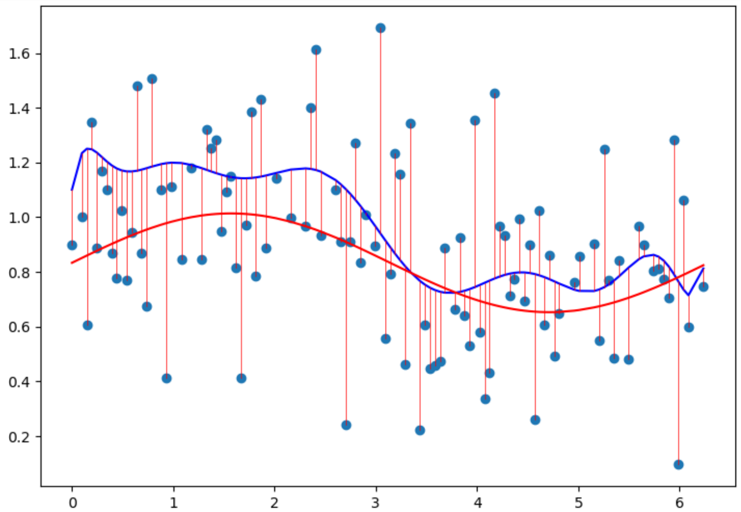

Figure 3: The mean of the absolute value of the weights for the blue model is 11.1.

Regularization

The solution to this problem is **regularization**.

Regularization comes in many forms, and we will cover a few in this course.

Regularization

The most common form is to penalize the weights from taking a high value.

This can be done by adding a penalization term to the loss function, which is then minimized by the optimizer.

Regularization

A simple term for the error is that of $L2$ regularization.

$$E(\mathbf{w}) = \frac{1}{2}||\mathbf{w}||^2 = \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

Regularization

Added to the sum-of-squares error for least squares, the final loss becomes

$$J(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{n}(h(\mathbf{x}_i; \mathbf{w}) - \mathbf{y}_i)^2 + \frac{1}{2}\mathbf{w}^T\mathbf{w}.$$

Regularization

This choice of regularization is beneficial in that it can be optimized via stochastic gradient descent.

Its form also allows it to be minimized in closed form via the normal equations.

Regularization

Taking the gradient of $J(\mathbf{w})$ above with respect to 0 and solving for $\mathbf{w}$ yields

$$\mathbf{w} = (\lambda I + \mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y},$$

where λ is a regularization hyperparameter.

Overfitting

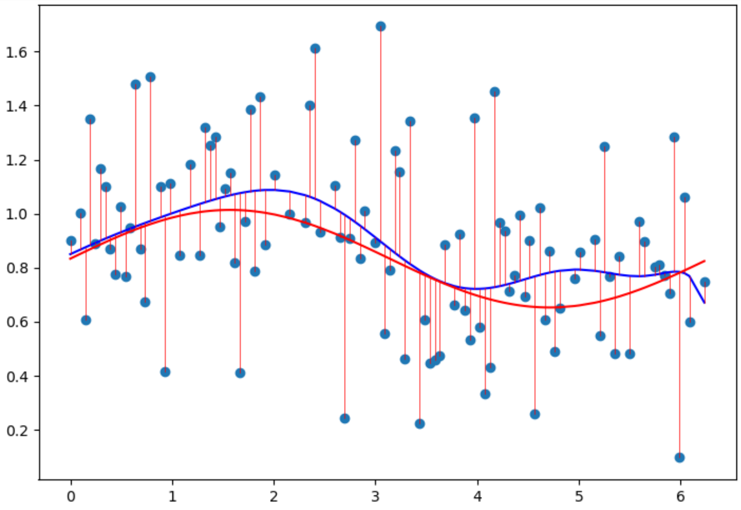

Figure 4: Least squares model fit with $L2$ regularization ($\lambda = 1$).

Regularization

Inspecting the weights as before, we can see that the mean of the absolute values of $\mathbf{w}$ is 0.0938.

Compared to the unregularized model with mean absolute weights of 11.1, regularization has done its job.

Regularization

Merely verifying that the weight values decreased is not good enough.

We expected this as the term was minimized via the loss function.

Evaluation on Test

To see which model generalizes better, we set aside some samples from the original dataset to use as testing.

With regularization, the model error on the test set is 1.8. Without regularization, the model error on the test set is 2.2.

Demo

Demo: Linear Regression