ECE 469/ECE 568 Machine Learning

Textbook:
Machine Learning: a Probabilistic Perspective by Kevin Patrick Murphy

Southern Illinois University

September 11, 2024

A recap for the last lecture

- In the last lecture, we discussed training, testing, and validation:
  - Splitting date-sets into training, testing, and validation
  - Hyperparameters
  - Overfitting vs. underfitting
  - Training, validation and testing errors
  - Generalization of machine learning models

Generalization and Capacity

Difference between optimization and machine learning

- When training a machine learning model, the training data set is used to learn a model, which minimizes training error. This is also a minimization problem.
- What separates machine learning from conventional optimization is that we also want the test error to be low and thus increasing generalization capacity.

Generalization and Capacity

- We can control whether a machine learning model is more likely to overfit or underfit by altering its "capacity".

Capacity

- A machine learning model's capacity is its ability to fit a wide variety of functions.
- Models with low capacity may struggle to fit the training set.
- Models with high capacity can overfit by memorizing properties of the training set that do not serve them well on the test set.

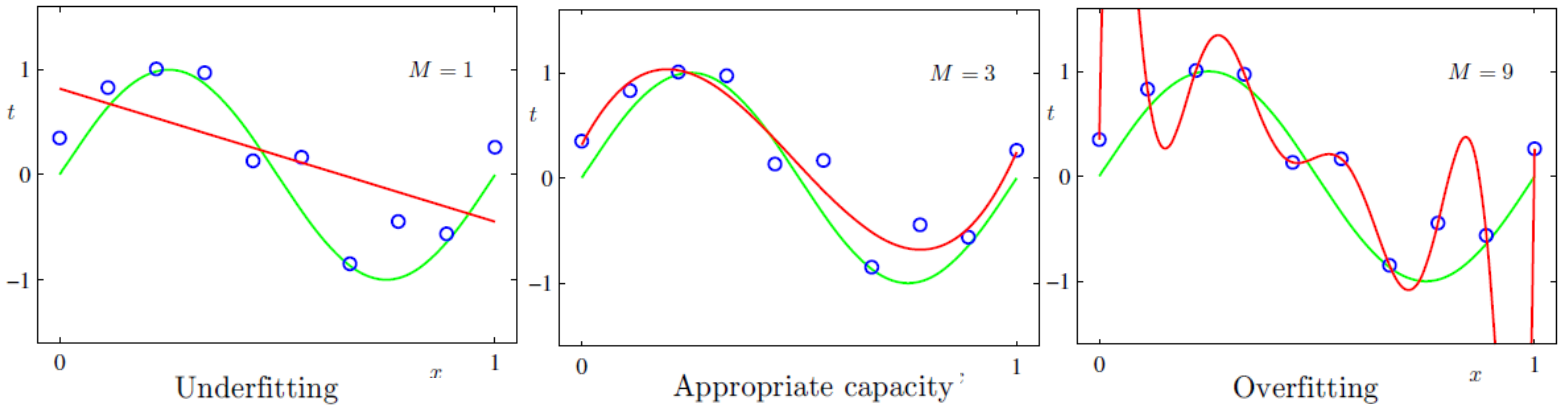
Generalization and Capacity

- We can control the capacity of a learning algorithm by choosing its hypothesis space, which is the set of functions that the learning algorithm is allowed to select as being the solution.
- Machine learning algorithms perform the best when their capacity is appropriate in regard to the true complexity of the task they need to perform and the amount of training data they are provided with.

Capacity trade-off

- Learning models with insufficient capacity are unable to solve complex tasks.
- Learning models with high capacity can solve complex tasks.
- However, when the capacity is necessarily higher, they may overfit.

Generalization and Capacity



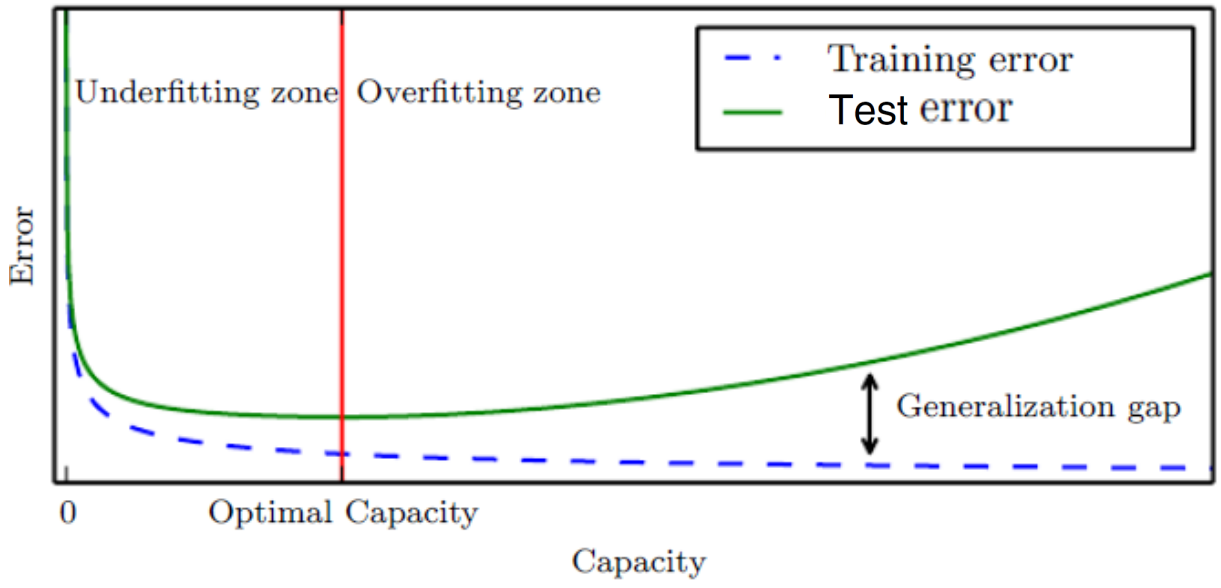
- The generalization and capacity has been known since 1200's.

Occam's razor

This principle states that among competing hypotheses that explain known observations equally well, one should choose the "simplest" one.

- We may trade-off "simplicity" for "accuracy of model fit".
- Moreover, if two models fit the data equally well, we may pick the simpler one.

Generalization and Capacity



- Generalization gap is the difference between the training error and test error.

The No-Free-Lunch Theorem

No-free-lunch theorem

This theorem states that averaged over all possible data generating distributions, every classification algorithm has the same average error rate when classifying previously unobserved points.

- Thus, no machine learning algorithm is universally any better than any other.
- The most advanced machine learning algorithm that one can conceive of has the same average performance (over all possible tasks).
- Note that this theorem is valid only when we average over all possible data generating distributions.
- Fortunately, one can make assumptions about the kinds of probability distributions we encounter in real-world applications.
- Then designing of machine learning algorithms that perform well on these distributions is practically viable.

Regularization in machine learning

- According to the no-free-lunch theorem, machine learning algorithms must be designed to perform well on a specific task.
- This can be accomplished by building a set of preferences into the underlying learning algorithm.
- Regularization is a technique that is used to increase the generalization capacity of machine learning models: –> To mitigate overfitting.

Regularization

- A learning algorithm can be give a preference for one solution in its hypothesis space over the other.
- Both functions are eligible, however one is preferred.
- The unpreferred solution can be chosen only if it fits the training data significantly better than the preferred solution.

Regularization in machine learning

Regularization: Mitigates overfitting

- Regularization can broadly be defined as any modification we make to a learning model/algorithm that is intended to reduce its test/generalization error.
- Regularization can be viewed as controlling a learning model's capacity than including or excluding members from the hypothesis space.
- Regularization is one of the main considerations of the field of machine learning.

Regularization in machine learning

- There are multiple ways to perform regularization.
- Based on the techniques used to overcome overfitting, we can classify the regularization techniques into three categories.
  - Modifications to the loss function
  - Modifications to sampling method for allocating data for training and validation
  - Modifications to training algorithm

Regularization - Modifications to loss function

- Here, we modify the original loss function to improve the generalization of machine learning models.
- Thus, one technique that can be used to control the overfitting phenomenon is regularization.
- Regularization involves adding a penalty term to the loss function in order to discourage the coefficients from reaching large values.
- The simplest such penalty term takes the form of a sum of squares of all of the coefficients. This is referred to as ridge regularization.

Regularization techniques under modifications to loss function:

- Ridge regression (l_2-regularization)
- Lasso regression (l_1-regularization)
- Elastic net regression
- Entropy regularization - will be discussed under classification

Regularization - Modifications to loss function

- A general "regularizer" can be defined as

$$\tilde{J}(\mathbf{w}) = J(\mathbf{w}) + \frac{\lambda}{N}\Omega(\mathbf{w}),$$

  where $\tilde{J}(\mathbf{w})$ is the modified loss function, $J(\mathbf{w})$ is the original loss function, $\Omega(\mathbf{w})$ is a regularizer, λ is a hyperparameter that serves as a control variable, and N is the dataset size.

- In general, a model that learns a function $f(x, \mathbf{w})$ can be regularized by adding a penalty called a regularizer to the loss function.

- For example, in ridge regression, the regularizer is $\Omega(\mathbf{w}) = ||\mathbf{w}||^2 = \mathbf{w}^T\mathbf{w}$.

- If the weight vector of the model is defined as $\mathbf{w} = [w_0, w_1, \cdots, w_n]^T$, then the ridge regularizer can be defined as

$$\Omega(\mathbf{w}) = ||\mathbf{w}||^2 = \mathbf{w}^T\mathbf{w} = w_0^2 + w_1^2 + \cdots + w_n^2 = \sum_{i=0}^{n} w_i^2.$$

Regularization - Modifications to sampling method

- These techniques can be used to overcome overfitting that results from limited data-set size.
- They aim to manipulate the available data-sets to create a fair representation of the actual input distribution.
- The regularization techniques that fall into this category include
  - Data augmentation
  - Various cross-validation techniques
- These techniques can be further discussed later in the course.

Regularization - Modifications to training algorithm

- These regularization techniques can be implemented by modifying the training algorithm in various ways.
- The regularization techniques that fall into this category include
  - Dropout
  - Early stopping
  - Injecting noise
- These techniques can be further discussed later in the course under deep-learning.

Regularization - Ridge regression

- Ridge regression is a regularized version of linear regression, and the loss function is modified by adding a quadratic function of weights.
- A control variable is also included and it serves as a hyperparameter.
- This modification forces the learning algorithm to not only fit the data but also keep the model weights as small as possible.
- The regularization term should only be added to the loss function during training.
- Once the model is trained, the unregularized loss function must be used to evaluate the model's performance.

Regularization - Ridge regression

- In ridge regularization, the loss function can be modified for linear regression to include weight decay.

$$\underbrace{\tilde{J}(\mathbf{w})}_{\text{modified loss function}} = \underbrace{J(\mathbf{w})}_{\text{original loss function}} + \underbrace{\frac{\lambda}{N}}_{\text{control variable}} \times \mathbf{w}^T\mathbf{w}$$

- Here, λ is a value chosen ahead of time that controls the strength of our preference for smaller weights, N is the training date-set size, and $\mathbf{w}^T\mathbf{w} = ||\mathbf{w}||^2 = w_0^2 + w_1^2 + \cdots + w_n^2$.

Regularization - Ridge regression

- Thus, ridge regularization leads us to a modified loss function of the form as follows:

$$\tilde{J}(\mathbf{w}) = \frac{1}{N}\sum_{i=0}^{N}(f(x_i, \mathbf{w}) - y_i)^2 + \frac{\lambda}{N}||\mathbf{w}||^2$$

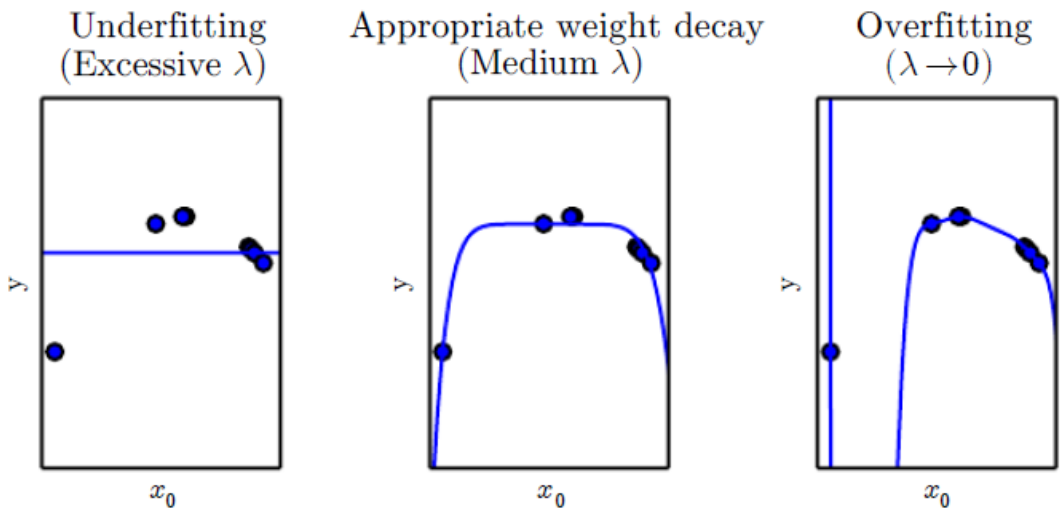
- Here, $||\mathbf{w}||^2 = \mathbf{w}^T\mathbf{w} = w_0^2 + w_1^2 + \cdots + w_n^2$ is called the l_2 norm of the weight vector $\mathbf{w}$.
- Alternatively, the modified loss function is given by

$$\tilde{J}(\mathbf{w}) = \frac{1}{N}\sum_{i=0}^{N}(f(x_i, \mathbf{w}) - y_i)^2 + \frac{\lambda}{N}\sum_{k=1}^{n} w_k^2$$

- The coefficient λ governs the relative importance of the regularization term compared with the sum-of-squares error term.
- Thus, λ is a hyperparameter of the model.

Regularization - Ridge regression

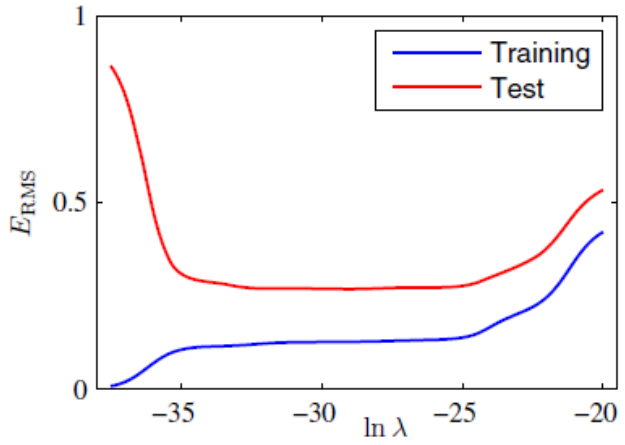
- For example, the impact of regularization can be shown by using a higher order ($M = 9$) polynomial instead of the true quadratic polynomial.
- Now, one can control a learning model's tendency to overfit or underfit through a weight.
- To this end, a high-degree polynomial regression model can be trained with different values of λ.



$\lambda \to 0$ implies that no preference is given.

A larger λ forces the weights to become smaller.

Regularization - Ridge regression



- The impact of the regularization term on the generalization gap can be observed by plotting the value of the root-mean square error for both training and test sets against $\log_e(\lambda) = \ln(\lambda)$.
- We observe that λ can control the capacity and effective complexity of the model.
- Eventually, it can determine the degree of over-fitting.

Regularization - Lasso regression

- In Lasso regularization, l_1 norm of the weight vector is added to the original loss function as

$$\tilde{J}(\mathbf{w}) = \frac{1}{N}\sum_{i=0}^{N}(f(x_i, \mathbf{w}) - y_i)^2 + \frac{\lambda}{N}\sum_{k=1}^{n}|w_k|$$

  where λ is a hyperparameter and $|w_k|$ is the absolute value of the weight w_k.

- An important characteristic of lasso regression is that it tends to eliminate the weights of the least important features.

- For example, it typically sets weights of the least important features to zero/closer-to-zero.

Regularization - Elastic Net Regression

- Elastic net regression is a middle ground between ridge regression and lasso regression that we discussed previously.
- This regularization term is a weighted sum of both ridge and lasso's regularization terms.
- The regularized loss function for the elastic net regression is given by

$$\tilde{J}(\mathbf{w}) = \frac{1}{N}\sum_{i=0}^{N}(f(x_i, \mathbf{w}) - y_i)^2 + r\left[\lambda\sum_{k=1}^{n}|w_k|\right] + (1-r)\left[\frac{\lambda}{N}\sum_{k=1}^{n}w_k^2\right]$$

- There is also another hyperparameter (r) to control the mix between Ridge and Lasso regularizers.
- When $r = 0$, Elastic net regression becomes Ridge regression.
- When $r = 1$, Elastic net regression becomes Lasso regression.

When to use regularization and which regularization technique?

- It is recommended to use one form of regularization to generalize your model. Plain linear regression must be avoided in learning models.
- Ridge regression is a good default.
- If there are only a few features that are useful, then Lasso or Elastic net regression is preferred as they tend to reduce the useless features' weights down to zero.
- Typically, Elastic net regression is preferred over Lasso regression as Lasso may sometimes behave erratically when the number of features is greater than the number of training instances or when several features are strongly correlated.