CS 316: Introduction to Deep Learning

Model Selection, Underfitting and Overfitting
Week 6

Dr Abdul Samad

Lecture Outline

- Generalized Models
- Training and Generalized Error
- Model Complexity
- Model Selection
- Validation Dataset
- Overfitting vs Underfitting

Generalized Models

- As machine learning scientists, our goal is to discover patterns.
- But how can we be sure that we have truly discovered a general pattern and not simply memorized our data?
- When we deploy the model in the future, we will encounter examples that the model has never seen before.
- Our predictions will only be useful if our model has truly discovered a general pattern.
- Our goal is to discover patterns that capture regularities in the underlying population from which our training set was drawn.

Training and Generalization Error

- The training error is the error of our model as calculated on the training dataset.
- Generalization error is the expectation of our model's error were we to apply it to an infinite stream of additional data examples drawn from the same underlying data distribution as our original sample.
- We can never calculate the generalization error exactly.
- In practice, we must estimate the generalization error by applying our model to an independent test set constituted of a random selection of data examples that were withheld from our training set.

Model Complexity

- When we have simple models and abundant data, we except the generalization error to resemble the training error.
- When we work with more complex models and fewer examples, we expect the training error to go down but the generalization gap to grow.
- Some of the factors that tend to influence the generalizability of a model class are as follows:
    - The number of tunable parameters. When the number of tunable parameters is large, models tend to be more susceptible to overfitting.
    - The values taken by the parameters. When weights take a wider range of values, models can be more susceptible to overfitting.
    - The number of training examples. It is trivially easy to overfit a dataset containing only one or two examples even if your model is simple. But overfitting a dataset with millions of examples requires an extremely flexible model.

Model Selection

- In machine learning, we usually select out final model after evaluating several candidate models. This process is called model selection.
- Sometimes models subject to comparison are fundamentally different in nature.
- At other times, we are comparing members of the same class of models that have been trained with different hyperparameter settings.
- In order to determine, the best among our candidate models, we will typically employ a validation dataset.

Validation Dataset

- In principle we should no touch our test set until after we have chosen all our hypermeters.
- If we overfit our training data, there is always the evaluation on the test data.
- But if we overfit the dataset, how would we ever know ?
- We should never rely on the test data for model selection. And yet we cannot solely rely on the training data for model selection either because we cannot estimate the generalization error on the very data, we use to train the model.
- In the real world, test data is discarded after just one use, and we can seldom afford a new test for each round of experiments.
- The common practice to address this problem is to split our data three ways, incorporating a validation dataset into addition to the training and test datasets.

K-Fold Cross-Validation

- When the training data is scare, we might not even to be able to afford to hold out enough data to constitute a proper validation set.
- One popular solution to this problem is to employ K-fold cross validation.
- The original training data is split into K non-overlapping subsets.
- Model training and validation are executed K times , each time training on K -1 subsets and validation on a different subset i.e., the one not used for training in that round.
- The training and validation errors are estimated by averaging over the results from the K experiments.

K-Fold Cross-Validation

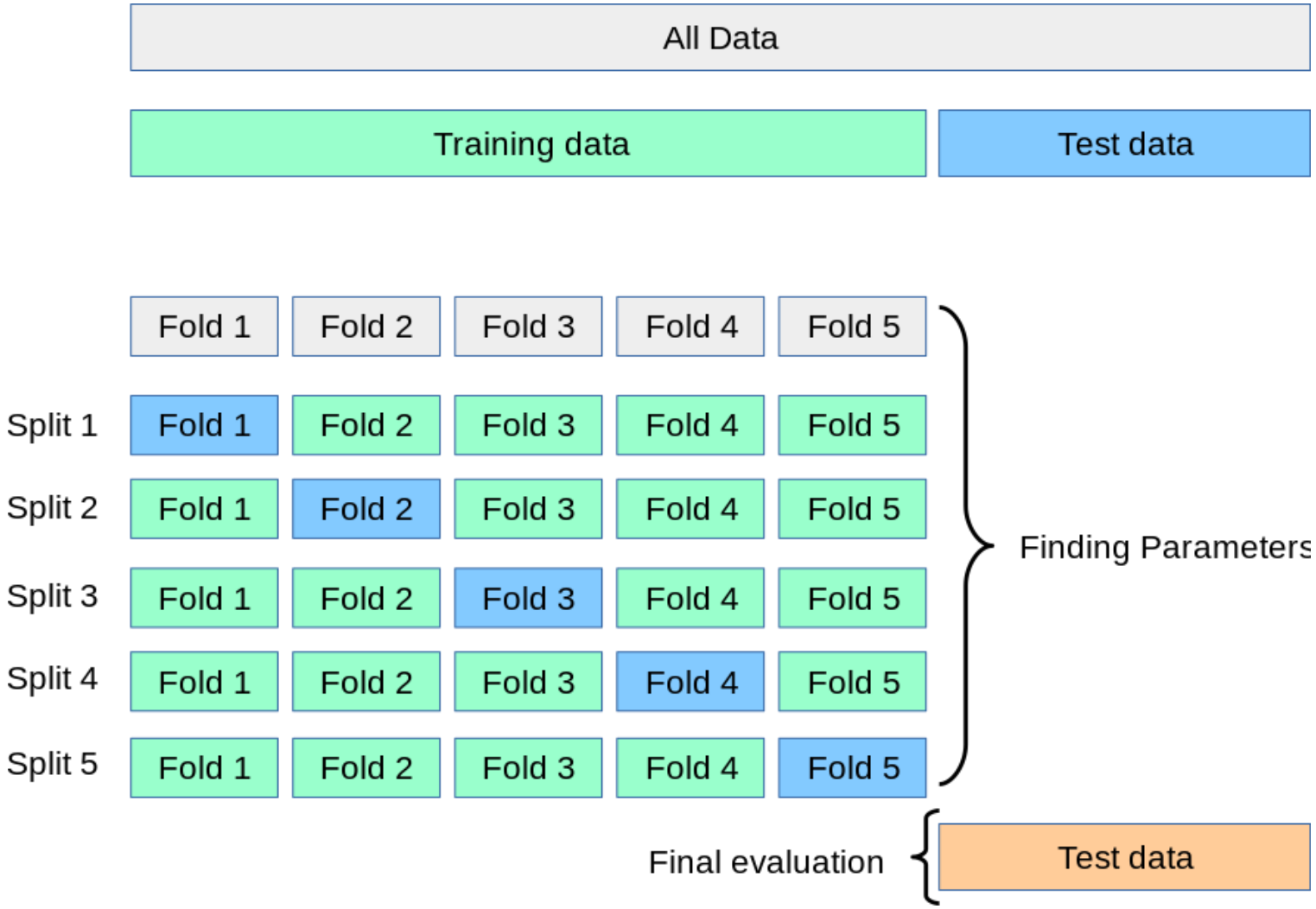

Figure Source

Overfitting and Underfitting

- When comparing training and validation errors, we must keep two common scenarios in mind: underfitting and overfitting.
- If the model cannot reduce training error, this may indicate that our model is too simple to capture the pattern we are attempting to model. Furthermore, because the difference in generalization between our training and validation errors is small, we could employ a more complex model. This is referred to as underfitting.
- On the other hand, overfitting occurs when our training error is significantly lower than our validation error.
- Whether we overfit or underfit can depend both on the complexity of our model and the size of the available training datasets.

Influence of model complexity on underfitting and overfitting

- A complex model has more parameters than a less complex model and the model function's selection range is wider.
- Fixing the training dataset, complex models should always lower (at worse, equal) training error relative to less complex models.
- The relationship between model complexity and underfitting vs. overfitting can be visualized as follows.

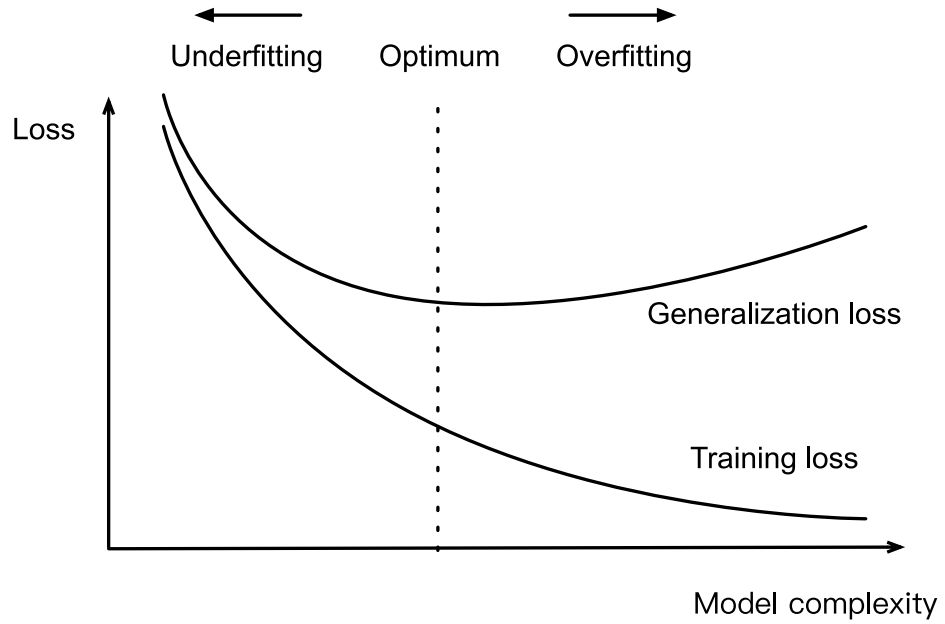

Figure Source

Influence of model complexity on underfitting and overfitting

- Fixing our model, the fewer samples we have in the training dataset, the more likely we are to encounter overfitting.
- As we increase the amount of training data, the generalization error typically decreases.
- Given more data, we might profitably attempt to fit a more complex model.
- In the absence of sufficient data, simpler models may become more difficult to beat.