Exam 2020

June 20, 2023

Exercise 0.1. Explain the difference between training error, validation error, test error, and generalization error.

- Training Error: This is the error that your model makes on the same data it was trained on. It's useful for diagnosing issues such as underfitting, but it's not a good measure of how well your model will perform on unseen data.
- Validation Error: This is the error that your model makes on a validation set, which is a separate set of data that the model wasn't trained on. It's used for tuning model parameters and choosing between different models.
- Test Error: This is the error that your model makes on a test set, which is another set of data that the model hasn't seen before. It's used to estimate how well the model will perform on unseen data.
- Generalization Error: This is a theoretical measure of how well your model will perform on new, unseen data. In practice, we can't calculate the generalization error exactly, so we estimate it using the test error.

Exercise 0.2. Explain what the Bayes error rate is and how it relates to the generalization error of any classifier.

The Bayes error rate is the lowest possible error rate that can be achieved by any classifier. It is determined by the overlap between the classes' distributions. In the case where the classes are perfectly separable, the Bayes error rate is zero. The difference between the Bayes error rate and the actual error rate of a classifier is the classifier's excess error. The generalization error of a classifier is an estimate of its excess error over the Bayes error rate.

Exercise 0.3. It is said that generative algorithms for supervised learning learn the joint distribution $p(x, y)$ where y is the target and x corresponds to a vector of explanatory variables, and discriminative algorithms learn $p(y|x)$. Please explain what this means.

Generative and discriminative models approach supervised learning differently:

- Generative Models: These models learn the joint probability distribution $p(x, y)$, where y is the target variable and x is a vector of explanatory variables. They model how the data is generated by learning the distribution of different classes. From this joint distribution, you can calculate the conditional distribution $p(y|x)$ which can be used to make predictions. Examples of generative models include Gaussian Naive Bayes, Linear Discriminant Analysis etc.
- Discriminative Models: These models learn the conditional probability distribution $p(y|x)$, which gives the distribution of the target variable given the explanatory variables. They focus on the boundary between classes rather than how the data of each class is distributed. Examples of discriminative models include Logistic Regression, Support Vector Machines etc.

Exercise 0.4. Please explain the difference between a parameter of a model and a hyper-parameter. You may use an example if you want.

A parameter and a hyperparameter of a model are both types of configurations that the model uses to make predictions, but they serve different roles:

- Parameters: These are the parts of the model that are learned from the training data. For example, in a linear regression model, the coefficients of the variables are parameters. They are found by fitting the model to the training data.

- Hyperparameters: These are settings on the model that are decided before the training process begins, and they are not learned from the data. For example, the learning rate in a gradient descent algorithm, the depth of a decision tree, or the number of hidden layers in a neural network are hyperparameters. They are typically set based on trial and error, prior knowledge, or through a process like cross-validation.

Exercise 0.5. Please explain the potential danger of not having any type of regularization in a modelling task and the danger of having too much of it.

Regularization is a technique used to prevent overfitting by adding a penalty term to the loss function that the model optimizes. This penalty discourages the model from assigning too much importance to any one feature, helping it to generalize better to unseen data.

- If there is no regularization, the model is at risk of overfitting to the training data. This means it will perform well on the training data, but poorly on unseen data because it has effectively memorized the training data rather than learning the underlying patterns.
- On the other hand, if too much regularization is applied, the model is at risk of underfitting. This means that it fails to capture important patterns in the data, leading to poor performance on both the training and unseen data.

Exercise 0.6. Please explain the relation between the bias/variance tradeoff and the k of the k-nearest neighbor algorithm.

The bias-variance tradeoff is a fundamental concept in machine learning which states that models with high complexity (low bias) tend to have high variance, while models with low complexity (high bias) tend to have low variance.

In the context of the k-nearest neighbor (k-NN) algorithm, 'k' is a hyperparameter that determines the number of neighbors to consider when making a prediction. A small 'k' value leads to a high complexity model (low bias, high variance), as it's more sensitive to noise in the data. On the other hand, a large 'k' results in a model with lower complexity (high bias, low variance), as it's more resilient to noise, but may oversimplify the patterns in the data. Balancing this tradeoff is key to achieving good performance with k-NN.

Exercise 0.7. What is the main objective of the resampling techniques that we have seen during the course (e.g. cross-validation)?

The main objective of resampling techniques, like cross-validation, is to estimate the performance of a model on unseen data. This is done by dividing the data into subsets: a training set to fit the model, and a validation set to evaluate it. This approach helps in detecting overfitting, where a model performs well on training data but poorly on unseen data. Additionally, cross-validation can be used to tune hyperparameters, by finding the values that give the best performance on the validation set.

Exercise 0.8. Can you think of a situation where the EM algorithm for clustering is preferable to k-means?

The Expectation-Maximization (EM) algorithm for clustering can be preferable to k-means in situations where the clusters are not spherical, or when the clusters have different variances. K-means assumes that all clusters are spherical and have similar variances, which can lead to poor performance if these assumptions are violated. EM, on the other hand, is a more flexible method that can handle clusters of different shapes and sizes. Furthermore, EM allows for "soft" assignment of data points to clusters, reflecting uncertainty in the assignments, whereas k-means only allows for "hard" assignment of data points to the closest cluster.

Exercise 0.9. What is the main purpose of the backpropagation algorithm in the context of neural networks?

The main purpose of the backpropagation algorithm in the context of neural networks is to efficiently compute the gradient of the loss function with respect to the weights of the network. This gradient is then used in optimization algorithms, such as stochastic gradient descent, to update the weights of the network and minimize the loss function. The key idea behind backpropagation is the chain rule of calculus, which allows for the gradient of the loss to be computed layer by layer, from the output layer back to the input layer. This makes training deep neural networks computationally feasible.