Machine Learning

Assignment 1 (Descriptive questions)

Mohammad Rouintan
Student No: 400222042
Shahid Beheshti University

(Spring 1402)

Exercise 1:

Can gradient descent get stuck in a local minimum when training a logistic regression model? Why?

Solution:

No. Gradient Descent cannot get stuck in a local minimum when training a Logistic Regression model because the cost function of Logistic Regression model is ***convex***.

Exercise 2:

Suppose you are using polynomial regression. You plot the learning curves and you notice that there is a large gap between the training error and the validation error. What is happening? What are three ways to solve this?

Solution:

If the validation error is much higher than the training error, this is likely because the model is ***overfitting*** the training set. One way to imrpove an overfitting model is ***increasing*** the size of the training set. Another way is ***reducing*** the polynomial degree, a model with fewer degrees of freedom is less likely to overfit. Last way to try is ***regularizing*** the model. Either ***L2 (Ridge Regression)*** or ***L1 (Lasso Regression)*** are good choices.

Exercise 3:

Suppose you are using ridge regression and you notice that the training error and the validation error are almost equal and fairly high. Would you say that the model suffers from high bias or high variance? Should you increase the regularization hyperparameter α or reduce it?

Solution:

The model suffers from ***high bias***, because the training error and the validation error are almost equal and high that means the model is ***underfitting***. We should try reducing the regularization hyperparameter α.

Exercise 4:

Why would you want to use:

- Ridge regression instead of plain linear regression (i.e., without any regularization)?
- Lasso instead of ridge regression?
- Elastic net instead of lasso regression?

Solution:

- A model with some ***regularization*** typically performs better than a model without any regularization, so we should prefer ***Ridge Regression*** instead of plain ***Linear Regression***. Also, Ridge Regression prevents ***overfitting***.
- ***Lasso Regression***, which uses ***L1 norm regularization***, automatically tends to push the weights down to ***exactly zero*** and eliminates the weights of the least important features. Therefore performs ***feature selection***, which is good if you suspect that only a few features actually matter.
- ***Elastic Net Regression*** is preferred over Lasso Regression because Lasso Regression may behave erratically when there are more features than training instances or when several features are ***strongly correlated***.

Exercise 7:

Compare bootstrapping with cross-validation. In which conditions we should use bootstrapping?

Solution:

1. ***Cross-Validation :***
   Cross validation is a procedure for validating a model's performance, and it is done by splitting the training data into k parts. We assume that the $k-1$ parts is the training set and use the other part is our test set. We can repeat that k times differently holding out a different part of the data every time. Finally, we take the average of the k scores as our performance estimation. Cross validation can suffer from bias or variance. Increasing the number of splits, the variance will increase too and the bias will decrease. On the other hand, if we decrease the number of splits, the bias will increase and the variance will decrease.
2. ***Bootstrapping :***
   bootstrap resamples with replacement (and usually produces new "surrogate" data sets with the same number of cases as the original data set). Due to the drawing with replacement, a bootstrapped data set may contain multiple instances of the same original cases, and may completely omit other original cases.

In summary, Cross validation splits the available dataset to create multiple datasets, and Bootstrapping method uses the original dataset to create multiple datasets after resampling with replacement. Bootstrapping it is not as strong as Cross validation when it is used for model validation. Bootstrapping is used more for statistical tests, ensemble machine learning, and parameter estimation. In particular, the bootstrap is useful when there is no analytical form or an asymptotic theory to help estimate the distribution of the statistics of interest. This is because bootstrap methods can apply to most random quantities.

Exercise 8 (Extra Point):

Explain nested cross-validation and 5x2 cross-validation in detail and when we should use them?

Solution:

1. ***Nested Cross-Validation :***
   Nested cross-validation (CV) is often used to train a model in which hyperparameters also need to be optimized. Nested CV estimates the generalization error of the underlying model and its (hyper)parameter search. Choosing the parameters that maximize non-nested CV biases the model to the dataset, yielding an overly-optimistic score. Model selection without nested CV uses the same data to tune model parameters and evaluate model performance. Information may thus "leak" into the model and overfit the data. The magnitude of this effect is primarily dependent on the size of the dataset and the stability of the model.

2. ***5 $\times$ 2 Cross-Validation :***
   The 5x2cv paired t test is a procedure for comparing the performance of two models (classifiers or regressors) that was proposed by Dietterich to address shortcomings in other methods such as the resampled paired t test and the k-fold cross-validated paired t test.

   To explain how this method works, let's consider to estimator (e.g., classifiers) A and B. Further, we have a labeled dataset D. In the common hold-out method, we typically split the dataset into 2 parts: a training and a test set. In the 5x2cv paired t test, we repeat the splitting (50percent training and 50percent test data) 5 times.

Exercise 9 (Extra Point):

How can we compare different models using statistical significance tests?

Solution:

Statistical significance tests are designed to address this problem and quantify the likelihood of the samples of skill scores being observed given the assumption that they were drawn from the same distribution. If this assumption, or null hypothesis, is rejected, it suggests that the difference in skill scores is statistically significant. Although not foolproof, statistical hypothesis

testing can improve both your confidence in the interpretation and the presentation of results during model selection.
In the case of selecting models based on their estimated skill, we are interested to know whether there is a real or statistically significant difference between the two models.
If the result of the test suggests that there is insufficient evidence to reject the null hypothesis, then any observed difference in model skill is likely due to statistical chance. If the result of the test suggests that there is sufficient evidence to reject the null hypothesis, then any observed difference in model skill is likely due to a difference in the models. The results of the test are probabilistic, meaning, it is possible to correctly interpret the result and for the result to be wrong with a type I or type II error. Briefly, a false positive or false negative finding.
Comparing machine learning models via statistical significance tests imposes some expectations that in turn will impact the types of statistical tests that can be used; for example:
Skill Estimate. A specific measure of model skill must be chosen. This could be classification accuracy (a proportion) or mean absolute error (summary statistic) which will limit the type of tests that can be used. Repeated Estimates. A sample of skill scores is required in order to calculate statistics. The repeated training and testing of a given model on the same or different data will impact the type of test that can be used. Distribution of Estimates. The sample of skill score estimates will have a distribution, perhaps Gaussian or perhaps not. This will determine whether parametric or nonparametric tests can be used. Central Tendency. Model skill will often be described and compared using a summary statistic such as a mean or median, depending on the distribution of skill scores. The test may or may not take this directly into account.

Exercise 11:

Suppose the features in your training set have very different scales. Which algorithms (Gradient Descent, Normal Equation, SVD) might suffer from this, and how? What can you do about it?

Solution:

If the features in your training set have very different scales, the cost function will have the shape of an elongated bowl, so the ***Gradient Descent*** algorithms will take a long time to converge. To solve this you should scale the data before training the model. Note that the ***Normal Equation*** will work just fine without scaling.