Machine Learning

Debugging and Evaluation

Part One

Dr. Sherif Saad

Learning Objectives

Understand best practices for evaluating machine learning algorithms

Learn how to build good training sets

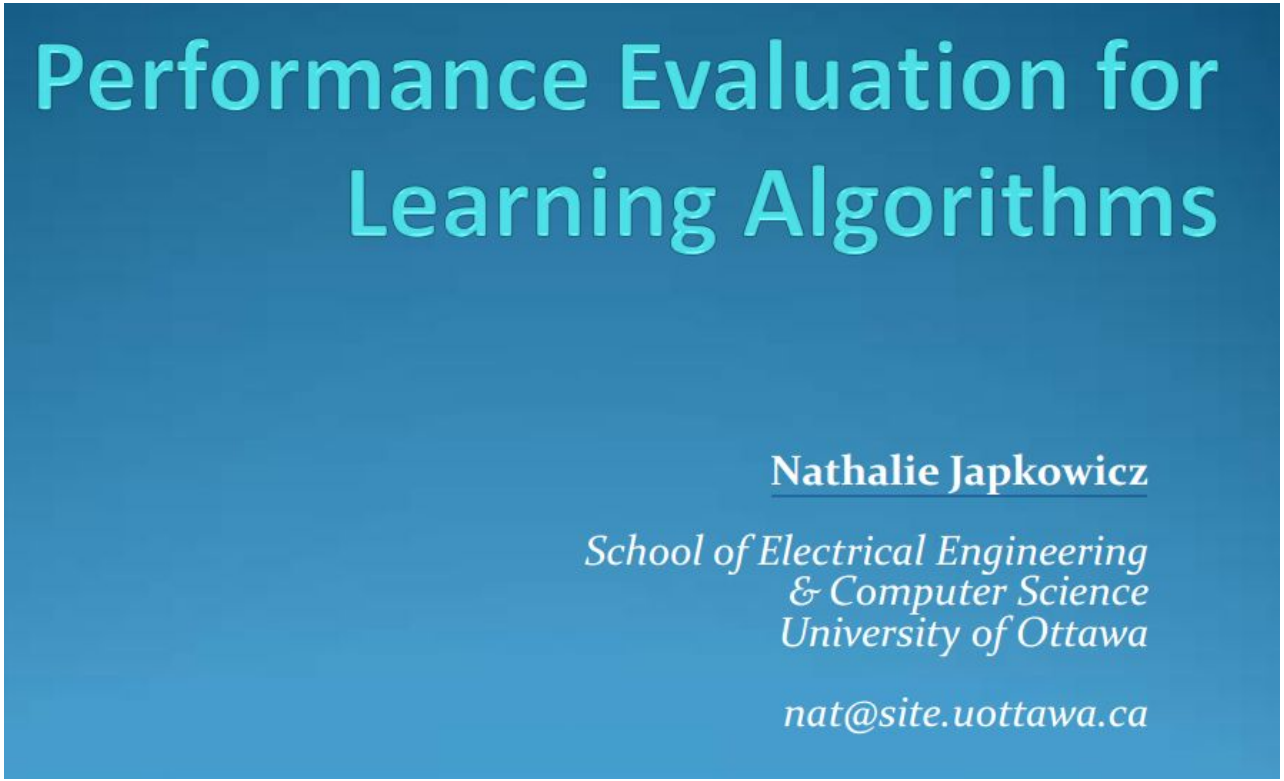
Diagnose the common problems of machine learning algorithms

Understanding and applying different performance metrics

Outlines

- PART ONE
  - Introduction to Machine Learning Evaluation
  - Prepare for Model Evaluation
  - Overfitting and Underfitting
- PART TWO
  - Debugging Machine Learning Algorithms
  - Performance Metrics

Problem Scope



Problem Scope

Motivation: My story

- A student and I designed a new algorithm for data that had been provided to us by the National Institute of Health (NIH).
- According to the standard evaluation practices in machine learning, we found our results to be significantly better than the state-of-the-art.
- The machine learning community agreed as we won a best paper award at ISMIS'2008 for this work.
- NIH disagreed and would not consider our algorithm because it was probably not truly better than the others.

Problem Scope

Motivation: My story

- A student and I designed a new algorithm for data that had been provided to us by the National Institute of Health (NIH).
- According to the standard evaluation practices in machine learning, we found our results to be significantly better than the state-of-the-art.
- The machine learning community agreed as we won a best paper award at ISMIS'2008 for this work.
- NIH disagreed and would not consider our algorithm because it was probably not truly better than the others.

Problem Scope

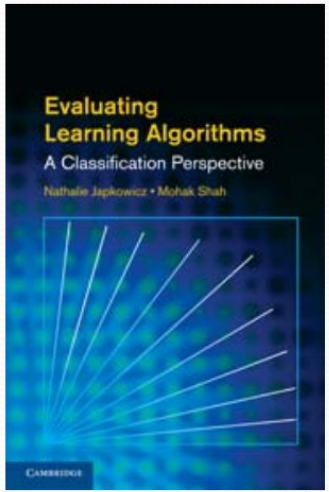
Motivation: My story (cont'd)

- My reactions were:
  - **Surprise:** Since my student and I properly applied the evaluation methodology that we had been taught and read about everywhere, how could our results be challenged?
  - **Embarrassment:** There is obviously much more to evaluation than what I have been told. How can I call myself a scientist and not know what the scientists of other fields know so well?
  - **Determination:** I needed to find out more about this and share it with my colleagues and students.

Problem Scope

Book Details

Evaluating Learning Algorithms:
A Classification Perspective
Nathalie Japkowicz & Mohak Shah
Cambridge University Press, 2011



- Review:

"This treasure-trove of a book covers the important topic of performance evaluation of machine learning algorithms in a very comprehensive and lucid fashion. As Japkowicz and Shah point out, performance evaluation is too often a formulaic affair in machine learning, with scant appreciation of the appropriateness of the evaluation methods used or the interpretation of the results obtained. This book makes significant steps in rectifying this situation by providing a reasoned catalogue of evaluation measures and methods, written specifically for a machine learning audience and accompanied by concrete machine learning examples and implementations in R. This is truly a book to be savoured by machine learning professionals, and required reading for Ph.D students."
Peter A. Flach, University of Bristol

What do we learn from this story?

As a researcher do not work with National Institute of Health (NIH)

Machine Learning Evaluation and Debugging is not a simple topic.

Machine Learning Evaluation and Debugging is divided into two main categories:

- Evaluation and Debugging of new generic and general purpose algorithm.
- Evaluation and Debugging of tuned and optimized (customized) existing machine learning algorithm (applied machine learning)

Evaluating and Debugging ML Algorithms

Many different machine learning algorithms are available.

Every algorithm has its inherent biases, and in general, no single algorithm is superior to others algorithms.

The selection of the machine learning algorithm is driven by the context of the problem and the data.

How do we select a machine learning algorithm, and how do we evaluate the selected machine learning algorithm?

Evaluating and Debugging ML Algorithms

It is essential to compare at least several different algorithms to train and select the best performing model.

It is important to decide which performance metric or measurement we will use to evaluate the selected models.

The classification accuracy is most commonly applied metric to evaluate classification algorithms, which is defined as the proportion of correctly classified observation.

Training & Testing Data

We want to be able to estimate the performance of the machine learning algorithm in production.

Randomly divide the available data into a separate training and test set.

We use the training set to train and tune the selected machine learning algorithms. We keep the test until the very end to evaluate the final version of the machine learning model.

How big the training and testing dataset? Are there any recommendations?

Training & Testing Data

In general, the training data is expected to be bigger than the testing data. Common settings are between (60% for training, 40% for testing) and (80% for training, 20% for testing).

In case of large datasets, 90:10 or 99:1 splits into training and test subsets are also common and appropriate.

How do we know which model performs well on the final test dataset and real-world data if we don't use this test set for the model selection but keep it for the final model evaluation? How can we estimate the generalization performance of the machine learning algorithm?

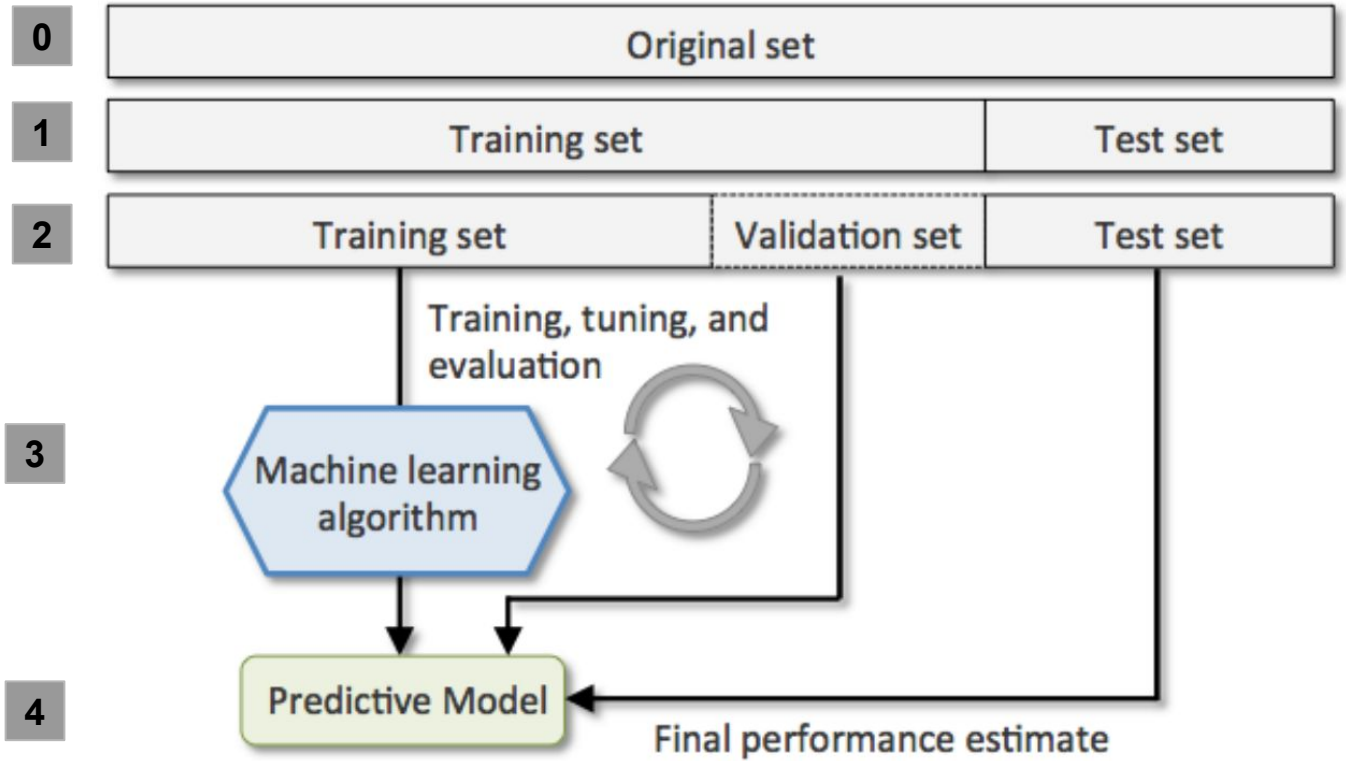
Training & Testing Data

Different cross-validation techniques can be used where the training dataset is further divided into training and validation subsets in order to estimate the generalization performance of the model.

Preparing Data Set for Machine Learning

1. Divide the available data into training set and test set
2. Divide the training set into training set and validation set
3. Use the two sets from step 2 to train, tune, and optimize your machine learning algorithm
4. Estimate the performance of the model in production using the test set.

Training & Testing Data



Machine Learning Model Selection

Do not expect that the default parameters of the different learning algorithms provided by off-the-shelf libraries (weka, scikit-learn, hadoop, etc) are optimal for our specific problem.

Model Selection, is the process of tuning and comparing different parameter settings of the machine learning model to improve the performance for making predictions on unseen data. Also known as finding the optimal hyperparameter values for the ML model.

If we simply use the test data over and over for model selection then it is not a test data anymore [WHY??]

Machine Learning Model Selection

Given a set of machine learning models M = {M1, M2, . . . , MR}. We want to select the model that is expected to best on unseen data.

The available models could be:

- Instances of same model with different settings or hyperparameters. For example KNN with different K options, Decision Trees with different depth and split settings.
- Completely different machine learning models. For example SVM, DTs, KNN, or NN.

Using K-fold Cross Validation

A disadvantage of splitting the data into three subsets (training validation and testing) is that the *performance estimate is sensitive to how we partition the training set into the training and validation subsets*. In general, different samples will give different estimation.

One solution to this problem is to apply a K-fold cross validation. In this case we randomly split the training dataset into k folds without replacement, where k −1 folds are used for the model training and one fold is used for validation (testing)

Using K-fold Cross Validation

K-fold cross-validation is a resampling technique without replacement, the advantage of this approach is that each sample point will be part of a training and test dataset exactly once.

10-fold cross-validation is the most common K-fold cross-validation settings.

The training data set is divided into 10 folds, and during the 10 iterations, 9 folds are used for training, and 1 fold will be used as the test set for the model evaluation. Also, the estimated performances $\boldsymbol{E_i}$ (for example, classification accuracy or error) for each fold are then used to calculate the estimated average performance E of the model

10-fold Cross Validation

K-fold Cross Validation

Selecting a good value of K (number of folds) is usually based on the size of the available data.

If we are working with relatively small training sets, it can be useful to increase the number of folds. if we are working with large datasets, we can choose a smaller value for k (e.g. k=5)

Leave-One-Out (LOO) cross-validation method. In LOO, we set the number of folds equal to the number of training samples (k = n) so that only one training sample is used for testing during each iteration.

Underfitting | Overfitting

Overfitting in this case the machine learning model performs well on training data but does not generalize well to unseen data (test data).

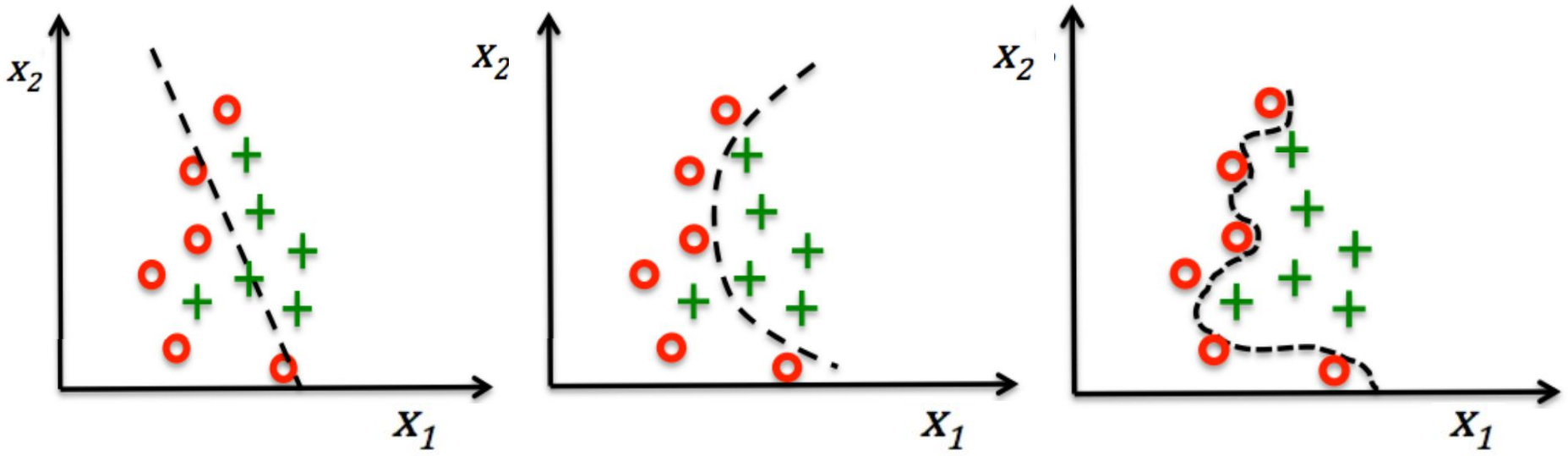
If a model suffers from overfitting, we also say that the model has a high variance. This is usually happen when we have too many parameters (features) that lead to a model that is too complex given the underlying data.

Underfitting on the other hand means our model is not complex enough (low learning ratio) to capture the patterns and the structure of the data. Will perform poorly on unseen data. Also known as high bias.

Usually we apply regularization methods to find a balance between overfitting and underfitting.

Underfitting | Overfitting

You have three models (black dashed lines) which one overfit the data, underfit the data and which one is a good balance between overfitting and underfitting



Questions