# 2 - A Few Useful things to know about Machine Learning

| | |
|---|---|
| Authors | Pedro Demingos |
| Implementation | No |
| Started On | @Jun 26, 2020 |
| Status | Completed |
| URL | https://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf |

# ABSTRACT

This paper aims to convey some of the "folk knowledge " that is need to successfully develop machine learning applications.

- This article mainly focuses on classification

# LEARNING = REPRESENTATION + EVALUATION + OPTIMIZATION

▼ ***Representation***

A classifier must be represented in some formal language that the computer can handle.

The set of learnable classifiers is called the ***hypothesis space*** of the learner.

How the input is represented is also an important point to address

- ***Evaluation***

  We need an evaluation/objective/scoring function to distinguish good classifiers from bad ones.

- ***Optimization***

  We also need a method to search among the classifiers for the highest scoring ones.

# IT'S GENERALIZATION THAT COUNTS

The fundamental aim of machine learning is to generalize beyond the training data.
So, we always should have a separate test data just for testing and evaluating the final model and also should be careful to not leak the test data into the training phase of the model during hyperparameter tuning.

In the early days, the need for a separate test data was not fully appreciated as the data was not that hard or complex to generalize. But in the modern era, the data that the model is trying to learn is becoming increasingly more complex.

# DATA ALONE IS NOT ENOUGH

***No Free Lunch Theorem*** by David Wolpert:

> No one model works best for every problem. The assumptions of a great model for one problem may not hold for another problem, so it is common in machine learning to try multiple models and find one that works best for a particular problem.

# OVERFITTING HAS MANY FACES

When the classifier outputs 100% accuracy on the training data and 50% accuracy on the testing data, when in fact, it could have achieved 75% accuracy on both sets of data, it is said to have overfit the training data.

The generalization error can be decomposed into ***bias*** and ***variance.***

Bias is a learner's tendency to consistently learn the same wrong thing.

Variance is the tendency to learn the same wrong things irrespective of the real signal.

The below images compares the concepts of bias and variance to throwing darts aiming towards a target.

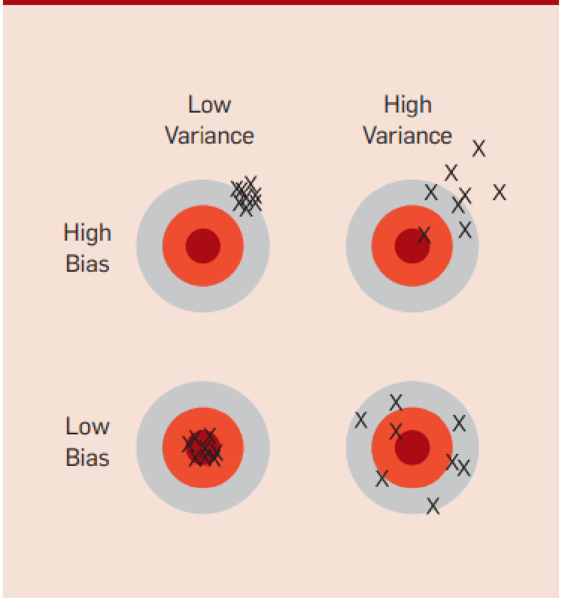

**Figure 1. Bias and variance in dart-throwing.**

***Some methods to combat overfitting:***

- Cross-validation can help to combat overfitting. But at the same time, too much hyperparameter tuning can lead to overfitting.
- Regularization - This can penalize the models with more structure thus preventing them from overfitting.
- Perform a statistical significance test

  Example Given: Chi Square Test - a way to test whether a decision is just a random choice or not.

# INTUITION FAILS IN HIGH DIMENSIONS

The biggest problem in machine learning is the ***curse of dimensionality***.

Our intuitions, which mostly come from 1 to 3 dimensional perceptions does not often translate to higher dimensions.

So, gathering more and more features might bite back in the form of the curse of dimensionality.

## THEORETICAL GUARANTEES ARE NOT WHAT THEY SEEM

One of the major developments of recent decades has been the realization that we can have guarantees on the results of induction (process of inferring general rules from specific data).

If learner A is better than B given infinite data, B is often better than A given finite data.

The main role of theoretical guarantees in machine learning is not to make practical decisions but to drive forward algorithm design. Just because a learner has a theoretical justification and works well in practice doesn't mean the former is the reason for the latter.

## FEATURE ENGINEERING IS THE KEY

Learning is easy if you have many independent features that correlates well with the class. On the other hand, if the class is a very complex function of the features, they it would be significantly harder for the model to learn the data.

Often, the raw data is not readily suitable for learning, but you can construct more useful features from those features. This is called ***Feature Engineering*** and where most of the time goes in a machine learning project. This requires intuition and creativity.

Machine learning is not a one-shot process of gathering the data and training the model on the data. But rather, it is an iterative process of gathering data, training the model, evaluating the model, making changes to the data/model, and then rinse & repeat.

> 💡 Bear in mind that the features that look useless in isolation could be extremely useful in combination with other features

## MORE DATA BEATS A CLEVER ALGORITHM

A dumb algorithm with lots and lots of data beats a clever algorithm with modest amounts of it.

***Three bottlenecks to machine learning: time, memory and training data.***

In the older days, data was the major problem. But now, big data is easily available. But there is no time to actually process all this data.

***This leads to a paradox:*** Even though more data means that more complex classifiers can be learned, in practice simple classifiers wind up being used, because complex ones takes too long to learn.

Machine learning projects wind up having a significant component of learner design, and the practitioners need to have some expertise in it. In the end, the biggest bottleneck in machine learning turns out to be human cycles. In research papers, learners are typically compared on measures of accuracy and computational cost. But human effort saved and insight gathered, although harder to measure, are often more important.

# LEARN MANY MODELS, NOT JUST ONE

Creating ***ensemble models*** is now standard in machine learning.

***Ensemble learning*** is a technique in which multiple models (usually weak learners) are trained to solve the same problem and combined to get better results.

There are three main ways of combining these weak learners:

- ***Bagging:***

  learns the data independently from each other in parallel and combines them following some averaging process

- ***Boosting:***

  learns them sequentially (the base model depends on the previous ones) and combines them following a deterministic strategy

- ***Stacking:***

  learns them in parallel and combines them by training a meta-model to output a prediction based on the different weak models predictions

The trend is towards larger and larger ensemble models.

# SIMPLICITY DOES NOT IMPROVE ACCURACY

There is no necessary connection between the number of parameters of a model and its tendency to overfit.

The conclusion in this section is that simpler hypotheses should be preferred because simplicity is a virtue in its own right, not because of a connection with accuracy.

# REPRESENTABLE DOES NOT IMPLY LEARNABLE

If the hypothesis space has many local optima of the evaluation function, the learner might not find the true function even if it is representable.

Given finite data, time and memory, standard learners can learn only a tiny subset of all possible functions, and these subsets are different for learners with different representations. Therefore the key question is not “Can it be represented?” to which the answer is often trivial, but “Can it be learned?”.

Some representations are exponentially more compact than others. As a result, they may also required exponentially less data to learn those functions.

# CORRELATION DOES NOT IMPLY CAUSATION

Machine learning is usually applied to observational data, where the predictive variables are not under the control of the learner, as opposed to experimental data, where they are.

Correlation is a sign of a potential causal connection, and it is a guide to further investigation

# CONCLUSION

Like any discipline, machine learning has a lot of “folk wisdom” that can be difficult to come by, but is crucial for success. This article summarized some of the most salient items.