# SCRUTINY OF CODIFICATION AND RETROGRESSION IN MACHINE LEARNING

**A.Poornima**
*Department of Computer Science, Department of Computer Application*
*Marudhar Kesari Jain College for women, Vaniyambadi, Vellore, India*

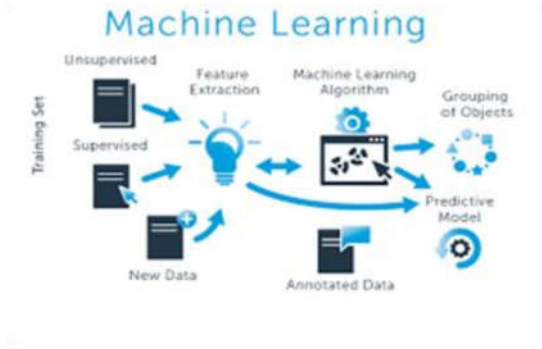
**M.Prithi**
*Department of Computer Science, Department of Computer Application*
*Marudhar Kesari Jain College for women, Vaniyambadi, Vellore, India*

## Abstract

*Service functionality can be provided by more than one service consumer. In order to choose the service which creates the most benefit before its consumption, a selection based on previous measurable experiences by other consumers is beneficial. In this paper, we present the results of our analysis of two machine learning approaches to predict the best service within this selection problem. The first approach focuses on classification, predicting the best performing service, while the second approach focuses on regression, predicting service performances which can then be used for the determination of the best candidate. We assessed and compared both approaches for service recommendation*

## Introduction

Machine learning is behind some of the coolest technological innovations today, Contrary to popular perception, however, you don't need to be a math genius to successfully apply machine learning. As a data scientist facing any real-world problem, you first need to identify whether machine learning can provide an appropriate solution. First, we will think how to determine which of the four basic approaches you'll take to solve the problem: classification, regression, clustering or recommendation.




ML is actually a lot of things. The field is quite vast and is expanding rapidly, being continually partitioned and sub-partitioned ad nauseam into different sub-specialties and types of machine learning.

There are some basic common threads, however, and the overarching theme is best summed up by this oft-quoted statement made by Arthur Samuel way back in 1959: "[Machine Learning is the] field of study that gives computers the ability to learn without being explicitly programmed."

"A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E." -- Tom Mitchell, Carnegie Mellon University

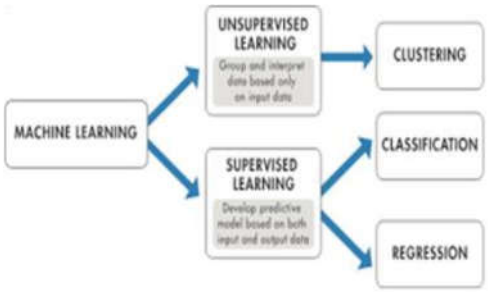
So if you want your program to predict, for example, traffic patterns at a busy intersection (task T), you can run it through a machine learning algorithm with data about past traffic patterns (experience E) and, if it has successfully "learned", it will then do better at predicting future traffic patterns (performance measure P).

The highly complex nature of many real-world problems, though, often means that inventing specialized algorithms that will solve them perfectly every time is impractical, if not impossible. Examples of machine learning problems include, "Is this cancer?", "What is the market value of this house?", "Which of these people are good friends with each other?",

ML solves problems that cannot be solved by numerical means alone.

Among the different types of ML tasks, a crucial distinction is drawn between supervised and unsupervised learning:

- **Supervised machine learning:** The program is "trained" on a pre-defined set of "training examples", which then facilitate its ability to reach an accurate conclusion when given new data.
- **Unsupervised machine learning:** The program is given a bunch of data and must find patterns and relationships.




**Overview of Related Work**

**Supervised Learning**

In supervised learning, the computer is provided with example inputs that are labeled with their desired outputs. The purpose of this method is for the algorithm to be able to "learn" by comparing its actual output with the "taught" outputs to find errors, and modify the model accordingly. Supervised learning therefore uses patterns to predict label values on additional unlabeled data.

For example, with supervised learning, an algorithm may be fed data with images of sharks labeled as fish and images of oceans labeled as water. By being trained on this data, the supervised learning algorithm should be able to later identify unlabeled shark images as fish and unlabeled ocean images as water.

A common use case of supervised learning is to use historical data to predict statistically likely future events. It may use historical stock market information to anticipate upcoming fluctuations, or be employed to filter out spam emails. In supervised learning, tagged photos of dogs can be used as input data to classify untagged photos of dogs.

**Unsupervised Learning**

In unsupervised learning, data is unlabeled, so the learning algorithm is left to find commonalities among its input data. As unlabeled data are more abundant than labeled data, machine learning methods that facilitate unsupervised learning are particularly valuable.

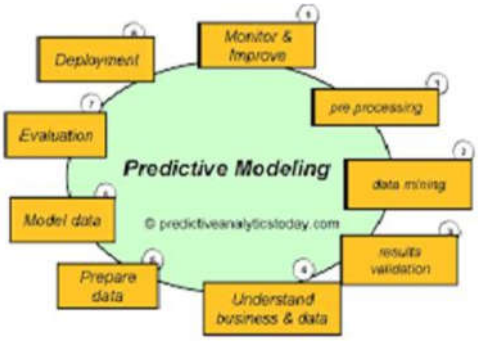
The goal of unsupervised learning may be as straightforward as discovering hidden patterns within a dataset, but it may also have a goal of feature learning, which allows the computational machine to automatically discover the representations that are needed to classify raw data.

Unsupervised learning is commonly used for transactional data. You may have a large dataset of customers and their purchases, but as a human you will likely not be able to make sense of what similar attributes can be drawn from customer profiles and their types of purchases. With this data fed into an unsupervised learning algorithm, it may be determined that women of a certain age range who buy unscented soaps are likely to be pregnant, and therefore a marketing campaign related to pregnancy and baby products can be targeted to this audience in order to increase their number of purchases.

Without being told a "correct" answer, unsupervised learning methods can look at complex data that is more expansive and seemingly unrelated in order to organize it in potentially meaningful ways. Unsupervised learning is often used for anomaly detection including for fraudulent credit card purchases, and recommender systems that recommend what products to buy next. In unsupervised learning, untagged photos of dogs can be used as input data for the algorithm to find likenesses and classify dog photos together.

**Predictive Modeling**

Predictive modeling is a process that uses data mining and probability to forecast outcomes. Each model is made up of a number of predictors, which are variables that are likely to influence future results. Once data has been collected for relevant predictors, a statistical model is formulated. The model may employ a simple linear equation, or it may be a complex neural network, mapped out by sophisticated software. As additional data becomes available, the statistical analysis model is validated or revised.




**Classification Predictive Modeling**

Classification algorithms are used when the desired output is a discrete label. In other words, they're helpful when the answer to your question about your business falls under a finite set of possible outcomes. Many use cases, such as determining whether an email is spam or not, have only two possible outcomes. This is called binary classification.

Multi-label classification captures everything else, and is useful for customer segmentation, audio and image categorization, and text analysis for mining customer sentiment. If these are the questions you're hoping to

answer with machine learning in your business, consider algorithms like naive Bayes, decision trees, logistic regression, kernel approximation, and K-nearest neighbors. Classification predictive modeling is the task of approximating a mapping function (f) from input variables (X) to discrete output variables (y).

The output variables are often called labels or categories. The mapping function predicts the class or category for a given observation. For example, an email of text can be classified as belonging to one of two classes: "spam" and "not spam".

- A classification problem requires that examples be classified into one of two or more classes.
- A classification can have real-valued or discrete input variables.
- A problem with two classes is often called a two-class or binary classification problem.
- A problem with more than two classes is often called a multi-class classification problem.
- A problem where an example is assigned multiple classes is called a multi-label classification problem.

It is common for classification models to predict a continuous value as the probability of a given example belonging to each output class. The probabilities can be interpreted as the likelihood or confidence of a given example belonging to each class. A predicted probability can be converted into a class value by selecting the class label that has the highest probability.

There are many ways to estimate the skill of a classification predictive model, but perhaps the most common is to calculate the classification accuracy. The classification accuracy is the percentage of correctly classified examples out of all predictions made.

For example, if a classification predictive model made 5 predictions and 3 of them were correct and 2 of them were incorrect

| |
|---|
| 1 accuracy = correct predictions / total predictions * 100 |
| 2 accuracy = 3 / 5 * 100 |
| 3 accuracy = 60% |

An algorithm that is capable of learning a classification predictive model is called a classification algorithm.

**Classification Example**

Suppose from your past data ( train data ) you come to know that your best friend likes the above movies. Now one new movie ( test data ) released. Hopefully, you want to know your best friend like it or not. If you strongly confirmed about the chances of your friend like the move. You can take your friend to a movie this weekend.

If you clearly observe the problem it is just whether your friend like or not. Finding a solution to this type of problem is called as classification. This is because we are classifying the things to their belongings (yes or no, like or dislike). Keep in mind here we are forecasting target class (classification) and the other thing this classification belongs to supervised learning. This is because you are learning this from your train data.

In this case, the problem is a binary classification in which we have to predict whether output belongs to class 1 or class 2 (**class 1 : yes, class 2: no**). As we have discussed earlier we can use classification for predicting more classes too. Like (**Color Prediction: RED, GREEN, BLUE, YELLOW, And ORANGE**)

**Regression Predictive Modeling**

Regression predictive modeling is the task of approximating a mapping function (f) from input variables (X) to a continuous output variable (y).A continuous output variable is a real-value, such as an integer or floating point value. These are often quantities, such as amounts and sizes.

For example, a house may be predicted to sell for a specific dollar value, perhaps in the range of $100,000 to $200,000.

- A regression problem requires the prediction of a quantity.
- A regression can have real valued or discrete input variables.
- A problem with multiple input variables is often called a multivariate regression problem.
- A regression problem where input variables are ordered by time is called a time series forecasting problem.

Because a regression predictive model predicts a quantity, the skill of the model must be reported as an error in those predictions.

There are many ways to estimate the skill of a regression predictive model, but perhaps the most common is to calculate the root mean squared error, abbreviated by the acronym RMSE.

For example, if a regression predictive model made 2 predictions, one of 1.5 where the expected value is 1.0 and another of 3.3 and the expected value is 3.0, then the RMSE would be:

| |
|---|
| 1 RMSE = sqrt(average(error^2)) |
| 2 RMSE = sqrt(((1.0 - 1.5)^2 + (3.0 - 3.3)^2) / 2) |
| 3 RMSE = sqrt((0.25 + 0.09) / 2) |
| 4 RMSE = sqrt(0.17) |
| 5 RMSE = 0.412 |

A benefit of RMSE is that the units of the error score are in the same units as the predicted value.

An algorithm that is capable of learning a regression predictive model is called a regression algorithm

**Regression Example**

Suppose from your past data (train data) you come to know that your best friend likes the above movies. You also know how many times each particular movie seen by your friend. Now one new movie (test data) released. Now your are going to find how many times this newly released movie will your friend watch. It could be 5 times, 6 times, 10 times etc…

If you clearly observe the problem is about finding the count, sometimes we can say this as predicting the value. Keep in mind, here we are forecasting a value (Prediction) and the other thing this prediction also belongs to Supervised learning. This is because you are learning this from you train data.

**Classification vs. Regression**

Classification predictive modeling problems are different from regression predictive modeling problems.

- Classification is the task of predicting a discrete class label.
- Regression is the task of predicting a continuous quantity.
- There is some overlap between the algorithms for classification and regression; for example:
- A classification algorithm may predict a continuous value, but the continuous value is in the form of a probability for a class label.
- A regression algorithm may predict a discrete value, but the discrete value in the form of an integer quantity.

Some algorithms can be used for both classification and regression with small modifications, such as decision trees and artificial neural networks. Some algorithms cannot, or cannot easily be used for both problem types, such as linear regression for regression predictive modeling and logistic regression for classification predictive modeling.

Importantly, the way that we evaluate classification and regression predictions varies and does not overlap, for example:

- Classification predictions can be evaluated using accuracy, whereas regression predictions cannot.
- Regression predictions can be evaluated using root mean squared error, whereas classification predictions cannot.

**Convert Between Classification and Regression Problems**

In some cases, it is possible to convert a regression problem to a classification problem. For example, the quantity to be predicted could be converted into discrete buckets.

For example, amounts in a continuous range between $0 and $100 could be converted into 2 buckets:

- Class 0: $0 to $49
- Class 1: $50 to $100

This is often called discretization and the resulting output variable is a classification where the labels have an ordered relationship (called ordinal).

In some cases, a classification problem can be converted to a regression problem. For example, a label can be converted into a continuous range.

Some algorithms do this already by predicting a probability for each class that in turn could be scaled to a specific range:

```
1 quantity = min + probability * range
```

Alternately, class values can be ordered and mapped to a continuous range:

- $0 to $49 for Class 1
- $50 to $100 for Class 2

If the class labels in the classification problem do not have a natural ordinal relationship, the conversion from classification to regression may result in surprising or poor performance as the model may learn a false or non-existent mapping from inputs to the continuous output range.

**Summary**

"Some algorithms have the word "regression" in their name, such as linear regression and logistic regression, which can make things confusing because linear regression is a regression algorithm whereas logistic regression is a classification algorithm"

- That predictive modeling is about the problem of learning a mapping function from inputs to outputs called function approximation.
- That classification is the problem of predicting a discrete class label output for an example.

- That regression is the problem of predicting a continuous quantity output for an example.
- Both the classification and regression concept is too important for prediction in machine learning.

**References**


1. https://searchenterpriseai.techtarget.com/definition/predictive-modeling
2. https://searchenterpriseai.techtarget.com/definition/predictive-modeling
3. https://www.toptal.com/machine-learning/machine-learning-theory-an-introductory-primer
4. https://machinelearningmastery.com/classification-versus-regression-in-machine-learning/
5. https://math.stackexchange.com/questions/141381/regression-vs-classification
6. https://www3.nd.edu/~rjohns15/cse40647.sp14/www/content/lectures/15-16%20-%20Regression.pdf
7. https://machinelearningmastery.com/classification-versus-regression-in-machine-learning/
8. https://www.datascience.com/blog/regression-and-classification-machine-learning-algorithms
9. https://ieeexplore.ieee.org/document/7196537/
10. https://www.datascience.com/blog/regression-and-classification-machine-learning-algorithms
11. https://www.sciencedirect.com/science/article/pii/S0004370213001082