The Machine Learning Process

Learn the general structure of how to approach Machine Learning problems in a methodical way.

The Process

When people think of Machine Learning, they often think of a program that is taking in data and spitting out predictions and insights. The process of performing Machine Learning often requires many more steps before and after the *predictive analytics*.

We try to think of the Machine Learning process as:

1. Formulating a Question
2. Finding and Understanding the Data
3. Cleaning the Data and Feature Engineering
4. Choosing a Model
5. Tuning and Evaluating
6. Using the Model and Presenting Results

1. Formulating a Question

What is it that we want to find out? How will we reach the success criteria that we set?

Let's say we are performing machine learning for a high-traffic fast-casual restaurant chain, and our goal is to improve the customer experience. We can serve this goal in many ways. When we're thinking about creating a model, we have to narrow down to one measurable, specific task. For example, we might say we want to predict the wait times for customers' food orders within 2 minutes, so that we can give them an accurate time estimate.

2. Finding and Understanding the Data

Arguably the largest chunk of time in any machine learning process is finding the relevant data to help answer your question, and getting it into the format necessary for performing predictive analysis.

We know that for supervised learning, we need *labeled* datasets, or datasets that have clear labels of what their ground truth is. For an example like the restaurant wait time, this would mean we would need many examples of past orders, tagged with how long the wait time was. Maybe the restaurant already tracks this data, but we might need to augment the

data collection with a timer that starts when the customer orders, stops when the customer receives their food, and records that information.

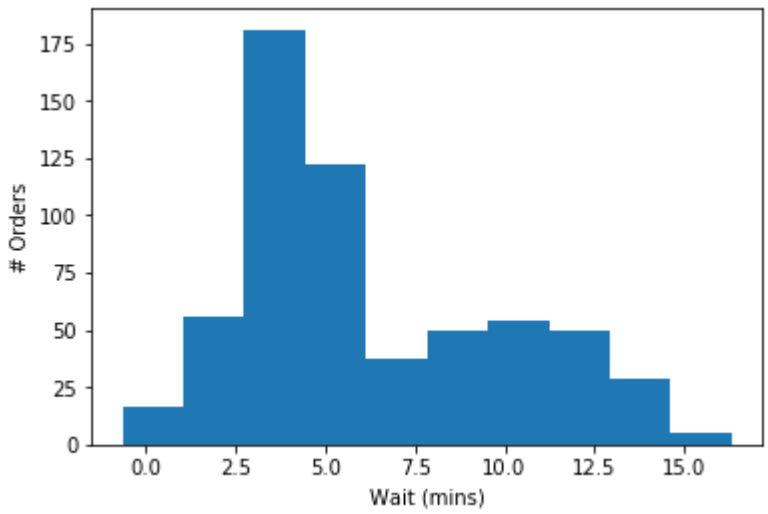
Creating this system of recording data, as well as gathering enough data to be able to train our model will take time.

Once you have your data, you want to understand it so that you will know what model to apply and what the outputs will mean. First, you will want to examine the summary statistics:

- Calculate means and medians to understand the distribution
- Calculate percentiles
- Find correlations that indicate relationships

You may also want to visualize the data, perhaps using box plots to identify outliers, histograms to show the basic structure of the data, and scatter plots to examine relationships between variables.

Let's say we're examining the existing distribution of wait times. We see that the overall average is 6.25 minutes per order. But we also produce this histogram:



We might glean from this that there are two main groups of orders. One group seems to cluster around 4 minutes, while another, smaller, group seems to cluster around 11 mins. We could use this to modify our question and build a model that will classify whether or not an order will be in this "short" timeframe, or in the "long" timeframe. Is it dependent on the food that it ordered? The time of day of the order?

Perhaps we just become aware of the bimodality of our data. If our model consistently predicts a wait time of around 6 or 7 minutes, then we are not taking into account the true structure of our data.

3. Cleaning the Data and Feature Engineering

Real data is messy! Data may have errors. Some columns may be empty. The features we're interested in might require string manipulation to extract. Cleaning the data refers to the process by which we address missing values and outliers, among other things that may affect our insights.

We may see that we have a group of orders that took over 20 minutes, due to an emergency in the kitchen one afternoon. This is pushing our average wait time up, and may skew our predictions. If we want to model the more general functioning of the restaurant, we may want to remove these values.

Feature Engineering refers to the process by which we choose the important features (or columns) to look at, and make the appropriate transformations to prepare our data for our model.

We might try:

- Normalizing or standardizing the data
- Augmenting the data by adding new columns
- Removing unnecessary columns

After we test our model on the data we have, we might go back and reengineer features to see if we get a better result.

4. Choosing a Model

Once we understand our dataset and know the problem we are trying to solve, we can begin to choose a model that will help us tackle our problem.

If we are attempting to find a continuous output, like predicting the number of minutes someone should wait for their order, we would use a regression algorithm.

If we are attempting to classify an input, like determining if an order will take under 5 minutes or over 10 mins, then we would use a classification algorithm.

The different classification and regression algorithms work better on different types of datasets. We use different models on categorical and numerical data, and different models on datasets with many features and datasets with few features. Our models also have

different levels of interpretability — how easy is it for us to see what these results mean and what led to them? When we teach the models, we will discuss the tradeoffs of using each one.

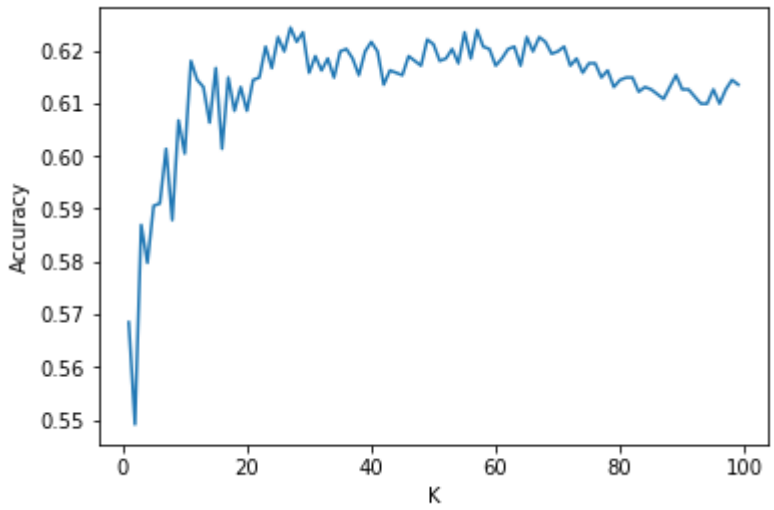
5. Tuning and Evaluating

We often want to set a metric of success, so that we know the model we've chosen is good enough. Are we looking for accuracy? Precision? Some combination of the two? We discuss this in our lesson on Precision and Accuracy.

Each model has a variety of parameters that change how it makes decisions. We can adjust these and compare the chosen evaluation metrics of the different variants to find the most accurate model.

For example, let's say we're using a K-Nearest Neighbors regression algorithm to solve the wait time prediction problem. This algorithm uses a parameter k, which you will learn about in the KNN lesson. We can adjust k to get different results.

Is it ideal to compare against 3 nearest neighbors? 10? 1? We can try many different values of k and see which one gives us the highest level of accuracy:



From this analysis, we would set our k to be 26, which got the highest level of accuracy.

6. Using the Model and Presenting Results

When you achieve the level of accuracy you want on your training set, you can use the model on the data you actually care about analyzing.

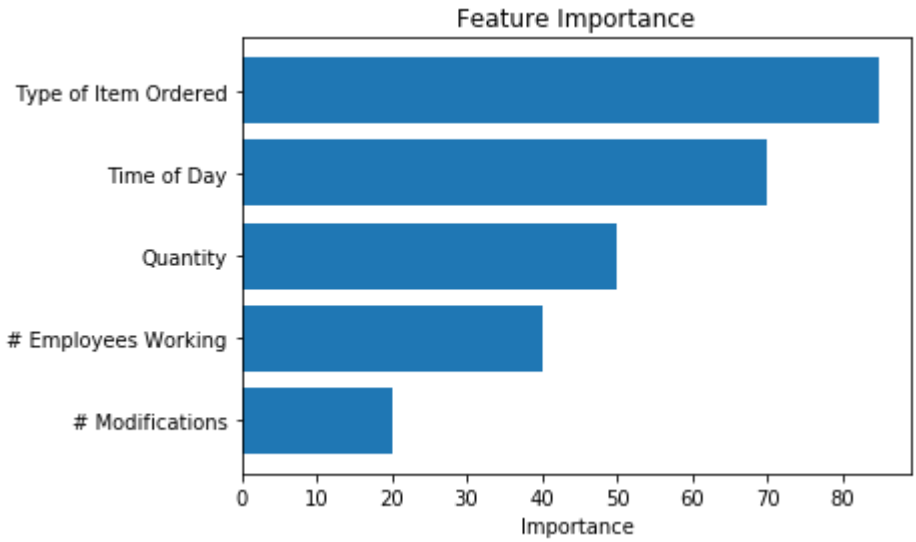
For our example, we can now start inputting new orders. The input could be an order, with features like:

- the type of item ordered
- the quantity
- the time of day
- the number of employees working

The output would be how long the order is expected to take. This information could be displayed to users.

An important step is being able to convey what you've learned and created, so that people can use it in the future.

Sometimes you learn more about your data by looking at the model. For example, using Multiple Learning Regression can give you insights into the importance of each feature. We can create a feature importance graph to visualize this for those unfamiliar with our model:



Your Process

The process we have outlined is a fairly standard process for performing machine learning. As you get experience going through this process on your own, with your own problems, you will start to form your own process. The steps may not be linear! As you clean your data, you may uncover a better question to ask. As you tune your model, you may realize you need more data, and go back to the collection step.

The important part is to stay curious, and to keep iterating until you find a model that works the best!