[1]

In this lecture we will consider one of the most important activities in machine learning that takes place long before we chose a learning algorithm or train our models. Data exploration is concerned with understanding our dataset, to gain deep insights into its content and suitability for using in a machine learning application

[2]

Recall that when we think of a dataset, we mean a set of unique observations (or instances) comprising descriptive features, continuous and categorical, and a target label. The instances can come from a variety of sources such as electronic sensors, human observations, survey results, data logs and so on. With such variety comes the risk that some instances may suffer from data quality issues such as missing values or invalid feature values through. When quality issues affect a significant proportion of the dataset, then this will inevitably impact the quality or even the viability of any machine learning model we train with this dataset. We assume for our purposes here that we are dealing with a supervised learning scenario which means that we have labelled instances by some means, a process that can also introduce errors into the dataset

[3]

It is recommended to start with a visual inspection of the dataset to get an initial impression of the data. This is an opportunity to see if the data is in a standard, structured, machine readable format such as comma-separated values or other popular serialization formats. If not, then this would be the first transformation step required. Next, look to see if each instance has approximately the same number of descriptive features. If you find a large variation here, then this might mean that the data is too unstructured to be usable in machine learning training. The visual inspection will also reveal how many missing values you might have to deal with later.

[4]

Assuming the initial visual inspection step passes, and the dataset is considered to be suitable to go forward, then it is useful at this point to generate a summary report. Most machine learning algorithms rely on probability theory and statistical inference so the summary report will detail some of the most important summary statistics about the dataset. Armed with this information, we can make decisions about which features look important, which we can use, which we may need to discard and any obvious problems with the data distribution. The report provides summary statistics about each descriptive feature and should treat continuous features and categorical features differently in this respect as we shall see. Regardless of the type of feature, we are interested in understanding the central tendency, variance and quality of the feature samples we have to work with.

[5]

For continuous features, we include the total count, cardinality, range, missing count and standard statistics for each continuous feature. The cardinality of a feature is the number of distinct values of the feature in our dataset and is important for inferring the role the feature values play. The first quartile value is the midpoint value between the minimum value and the median. The third quartile value is the midpoint value between the median and the maximum value. The missing value count is the number of instances for which the feature’s value is either not present or invalid.

[6]

For categorical features we also include the total count, cardinality and missing value count. But the summary statistics are slightly different and more appropriate for this kind of feature data. The mode value is the most frequently occurring categorical value for a given feature. The second mode value is the second most frequently occurring categorical value. We also note the frequency and percentage value for both mode and second mode. Again, the missing value count is the number of instances for which the feature’s value is either not present or invalid.

[7]

Another powerful tool for understanding the feature data is to identify the probability distributions of each. The simplest way to do this is by visualising the sample data points as a histogram distribution. Based on prior knowledge of the dataset, we may already have an idea of what kind of probability distributions we would expect to see for each feature. Plotting these now will confirm this or highlight potential problems with the data.

[8]

This analysis can also help us to filter out certain features that we know will not be helpful in model training. For example, a feature having a uniform distribution will almost certainly play no useful part and usually represents meta data items like data identifiers that we can safely discard.

[9]

On the other hand, it may be important to confirm that other features have, as expected, probability distributions like Gaussian, Bernoulli or Binomial. We can use histograms or density plots depending on the number of features and whether they are continuous or categorical.

[10]

The normal, or Gaussian, distribution is the most important probability distributions we deal with in statistics and machine learning. The probability density function as shown can be used to plot the density curve for the continuous feature values in our dataset. The curve is symmetric around the population mean (called mu), the peak value of the distribution. The height of the curve is dependent on the population standard deviation (called sigma). A standard distribution curve is plotted around a mean of zero. The normal distribution can be summarized by the 68-95-99.7 rule. This rule states that approximately 68% of the population values will lie one standard deviation from the mean (on both sides), 95% of the population values will lie two standard deviations from the mean and that 99.7% of the population values will lie three standard deviations from the mean. This implies that this an decreasing probability of observing values further from the mean in the normal distribution or, in other words, most of the population is relatively close to the population mean.

[11]

The next task is to visualize pairwise features with respect to each other. There are several reasons why his is useful. Firstly, we would desirably like to eliminate one half of any pairs of features which are strongly correlated. Strongly correlated features can impact on many algorithms’ performance. Having two features contributing to the same information and learning process is both redundant and potentially error prone, especially if they are negatively correlated. Another reason to eliminate redundancy in our data is that feature space reduction will allow our learning algorithms to train more efficiently by converging faster and using fewer computing resources. Let’s consider some data plots that we can use to help with this kind of pairwise analysis.

[12]

A scatter plot of two random variables, in this case two feature values, can visually reveal relationships between features. Specifically, a scatter plot matrix is the total set of pairwise plots of our dataset presented as one graph. It’s easy to see when two features are positively or negatively correlated though apparent linear relationships in the graphs. If we see any such suspected correlations, this would warrant further investigate and analysis of the impacted features such as calculating formal measures of this relationship.

[13]

Box plots of continuous features provide an assessible view of the central tendency of the feature values and the distribution skew. For example, the “age” feature in this box plot set clearly shows a skew in favour of younger ages in the population. A box plot is not useful for categorical features and can be ignored.

[14]

To complement the plot visualisations of potential feature correlations of continuous variables, we can compute the covariance and the correlation of the feature values in our dataset. Covariance is a function of the sample means of the variables in question and is defined as the weighted sum of the differences between the observed values for the feature pair and their respective population mean. A large negative value for covariance means that the features are likely negatively correlated. Similarly, a large positive value implies that they are positively correlated. Values near to zero imply no significant correlation.

[15]

The covariance measure only makes sense if the features being compared are measured in the same units. If different, then a better approach would be to use a normalized measure such as correlation. Correlation is a function of covariance and the standard deviation of the feature distributions of the variables in question. As with covariance, a negative result near -1 suggests a negative correlation, a positive result near 1 suggests a positive correlation and a result near zero implies no correlation or, in other words, the variables are said to be independent.

[16]

In summary, we have considered the important first step in machine leaning – getting to know the dataset. At the very least we need know whether the dataset is going to meet our needs at all and, if so, what transformations may be needed to prepare it for model training. By developing descriptive and visual statistical insights into the dataset feature values, we can see which features we need to keep, which ones we need to discard and which ones we need to fix. We can also make a better assessment of which of the possible training algorithms we should use to get the best results. The next step, following exploration, is to clean and prepare the data for training.