2. Classification

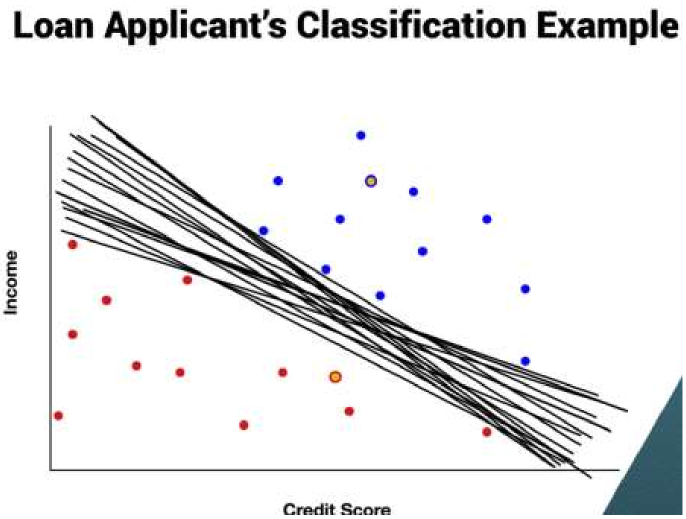
Monday, August 19, 2024 11:42 AM

Classification Introduction

Classification: the process of identifying and assigning categories to a collection of data, often into a binary category of "yes" or "no", thus differentiating them from one another.

The question becomes "what is the best way to differentiate them?" In the example below, there are an infinite number of ways to divide the blues (loan application accepted - not defaulted) from the reds (loan application denied - defaulted). What is the best line? The goal is to minimize the risk of classification errors.

Not all errors are created equal. We need to understand the tradeoff risks between each type of error; are the costs associated with giving a loan to a person who will default higher than the costs associated with not giving a loan to a person who wouldn't have defaulted? The more costly a type of error is, the more we want to move the dividing line away from that type of error.



Terminology

Data Tables: an organizational concept of data, where each row is an individual data point and each column is an attribute/feature/predictor/factor/covariate/variable. There may also be a special column called the response column which is the point of interest, such as if a customer repaid their loan.

Structured Data: data that can be stored in a structured way.

- Quantitative: age, credit score, sales, etc.
- Categorical: gender, eye color, etc. A special subset of categorical data is binary data.

Unstructured Data: data that is not easily described and stored, commonly text.

Unrelated Data: data where there is no relationship between data points, such as each row representing a different customer.

Time Series Data: data that is recorded over time, and thus the data points are related to one another (representing a change in time).

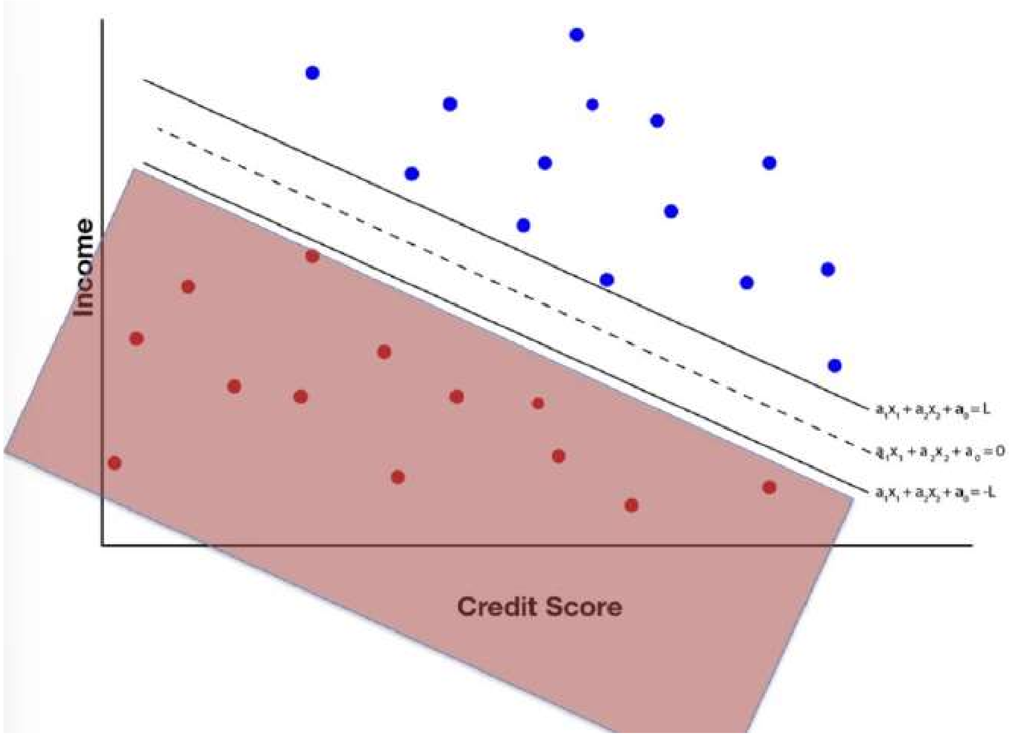
Classification: Support Vector Machines

Suppose we have a dataset with the below information:

- $n = number\ of\ attributes$
- $m = number\ of\ data\ points$
- $x_{ij} = jth\ attribute\ of\ ith\ data\ point$
- $y_i = response\ for\ data\ point\ i, where\ defaulted = -1\ and\ not\ defaulted = 1$

$$line: \sum_{j=i}^{n} a_j x_j + a_0 = 0$$

Thus the distance between the two solid lines is $\frac{2}{\sqrt{\sum_j a_j^2}}$ and we therefore want to minimize $\sum_{j=1}^{n} a_j^2$. In practical terms, this maximizes the margin between the red and blue sections of the graph.



There will likely still be points that are incorrectly classified. The error for these points is defined as the distance between that data point and the dashed line (which is the model). In mathematical terms:

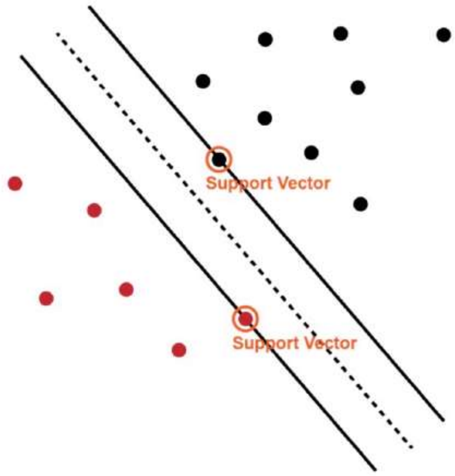
$$Error = \max\{0, 1 - (\sum_{j=1}^{n} a_j x_{ij} + a_0) y_i\}$$

If we subtract the margin from the error, we get:

$$minimize_{a_n} \sum_{i=1}^{m} \max\left\{0, 1 - \left(\sum_{j=1}^{n} a_j x_{ij} + a_0\right) y_i\right\} + \lambda \sum_{j=1}^{n} a_j^2$$

As λ grows larger, the importance of minimizing the margin grows. As λ shrinks towards 0, the importance of minimizing the error grows.

This approach to classification is called Support Vector Machine (SVM). The name comes from the two points (called support vectors) that maintain the shape of the margin.



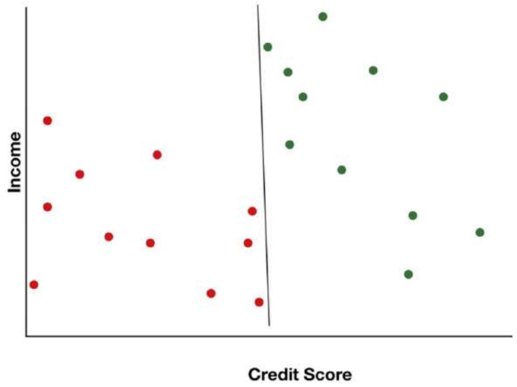
Once we have minimized this value and have our classifier, we can also consider how to handle the severity of certain errors. If we determine that the risks associated with giving a loan to a person who will default is twice the risk of not giving a loan to a person who would have paid it, we can shift the classifier around in a couple of different ways.

Note that the two solid lines have intercepts of -1 and +1 respectively in this example. This means that the lines can be shifted anywhere between [-1, 1] without making any mistakes of classification. Say that we determine the risks of approving a loan to a defaulter are twice as high as the risks of not giving a loan to a person who would pay it. Then:

- The intercept a_0 can be shifted: $\left[\frac{2}{3}(a_0 - 1) + \frac{1}{3}(a_0 + 1)\right]$
- Alternatively, we can add a multiplier m_j such that the there is a larger penalty (>1) for errors that are more costly: $minimize_{a_n} \sum_{i=1}^{m} m_j \max\{0, 1 - (\sum_{j=1}^{n} a_j x_{ij} + a_0) y_i\} + \lambda \sum_{j=1}^{n} a_j^2$

Additional Info:

- If one of the variables is of a wildly different scale than the other, it is necessary to scale the data. Credit score ranges from [0,1000], but income can range anywhere from 0 up to millions or more. Thus a small change in one variable can swamp a large change in the other.
- Additionally, if a variable's coefficient is near zero, then that variable is probably not relevant for classification, such as income in the below example.
- SVM classifiers do not have to be a straight line; kernel methods allow for nonlinear classifiers.
- SVM classifiers can work in multidimensions (with more attributes).
- Other methods like logistic regression can give probability answers (70% likely to default) rather than binary answers (deny loan application), which may be more appropriate for a given business question.



Scaling & Standardization

Scaling: a data preprocessing step that transforms the data such that every feature varies within the same range. Scaling is best used for models that need data to be within a bounded range, such as Neural Networks, certain Optimization models, or models using datasets that contain bounded data. Common linear scales include:

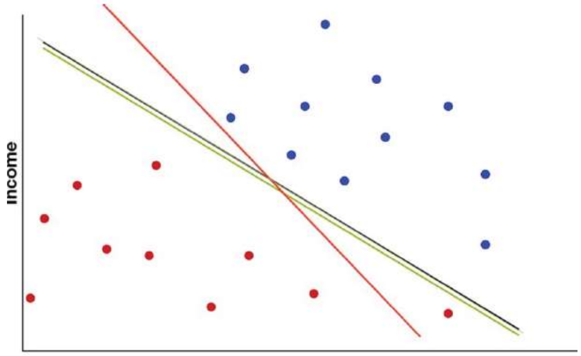
- Common: [0,1]
- Scaling factor by factor: $\mathrm{x}_{\mathrm{ij}}^{\mathrm{scaled}} = \frac{x_{ij} - \min DataPoint}{maxDataPoint - MinDataPoint}$
- Scaling between [a,b]: $x_{ij}^{scaled[a,b]} = x_{ij}^{scaled[0,1]}(a - b) + b$

Standardization: a data preprocessing step that transforms the data such that all the attributes are homogenized, not necessarily on a linear scale. Certain models perform better with standardization, such as Principal Component Analysis and Clustering. Common standardization scales include:

- Standard normal, with mean = 0 and standard deviation = 1: $\mathrm{x}_{\mathrm{ij}}^{\mathrm{standardized}} = \frac{x_{ij} - \mu_j}{\sigma_j}$

It is not always clear which method is better for a given model, so try both and see how the model performs.

With attributes that have wildly different scales, a small change in the variable with larger scale results in a slight change of the line (green) while a huge change in the variable with the smaller scale results in a very different line (red). This issue is prevented in a couple of different ways: scaling both variables down to the same interval or standardizing the data.

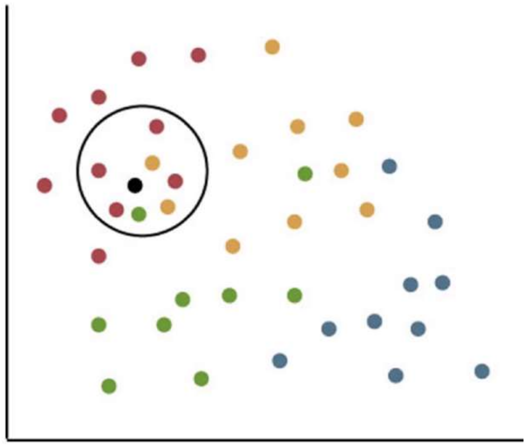


Classification: K-Nearest Neighbors

KNN is a simple model for dealing with classification problems that works well when there are more than two classes. The general algorithm for how KNN works is as follows:

1. Find the class of a new point

2. Pick the *k* points closest to this new point
3. The new point's class is the most common class among its neighbors



Important to note:

- There are multiple ways to measure distance.
- Attributes can be weighted by importance.
- Unimportant attributes can be removed.
- Finding the right value for *k* is a validation process by which several values are tried and the one with the best model output is selected.

Conclusion:

- Classification is the process by which data points are divided into groups based on similarity.
- This process has graphical intuition.
- Basic solution methods include machine learning algorithms SVM and KNN.

Lecture Slides

Module2_L1

Module2_L2

Module2_L3

Module2_L4

Module2_L5

Module2_L6

Module2_L7

Module2_L8

Module2_L7

Module2_L8