**Linear Models for Classification: Part 1**

**Basic concepts:**

* Supervised (with labels) vs unsupervised (label-less)
* Classification: All instances have known values for a set of features, called predictors/attributes, as well as a known or unknown class label. We aim to predict a class label.
* Regression: Similar to Classification, only that we want to predict a number instead of a class label. All features have a target attribute that we want to predict.

**Regarding features:**

* x are attributes (input features, predictor variables…) the info we find in the raw data.
* x’ are the features once they’ve been transformed and “engineered” (normalization, anonymization…)
* x’’ is the transformation of x’ once it’s been applied through the machine learning method. (e.g: deep neural networks etc.)

**Techniques:**

* KNN (*K-Nearest Neighbors*): we select the K nearest neighbors of a data point and based on their labels, the data point obtains the label of the upmost frequency. (near neighbors tend to have the same labels). We need a distance function to define the relationship between data points.

Instance space in KNN:

Gráfico, Gráfico de dispersión

Descripción generada automáticamente

* Perceptron: has a series of inputs affected by weights (*w0 + wi \* xi*).

An activation function is used to output the result of all inputs. It’s a type of neural network with no hidden layers and one output neuron.

Un dibujo de una cara feliz

Descripción generada automáticamente con confianza media

The decision regions defined by a perceptron are always separated by a linear hyperplane. Therefore, it cannot reproduce the classifications given by e.g XOR.

Gráfico

Descripción generada automáticamente

* Naïve Bayes Model: a Bayesian network where we have the class label Y with Xi, the different observed features. It is also a linear classifier.

Diagrama

Descripción generada automáticamente

There are more powerful types of classifiers: Support Vector Machines, Neural networks with hidden layers, Tree-augmented Naïve Bayes, General Bayesian Network models, among others…

**Overfitting**:

Regarding KNN, 1-Nearest Neighbor will always overfit as the training accuracy will always be 100%, by definition. Meanwhile the testing accuracy is lower than using 5-Nearest Neighbor, which actually obtains a worse training accuracy than the former model. Therefore, aiming for the best training accuracy isn’t the best option, especially when the testing accuracy goes down.

“Given a hypothesis space H, a hypothesis h ∈ H is said to overfit the training data if there exists some alternative hypothesis h’ ∈ H, such that h has smaller error than h’ over the training examples, but h’ has a smaller error than h over the entire distribution of instances”

Imagen que contiene Gráfico

Descripción generada automáticamente

In this example, hypothesis h overfits since it obtains a higher error than h’ in the test data, even though it performs better on the training data.

**Linear Functions**:

A linear function is defined by: *y(x1, …, xD) = w0 + w1x1 + … + wDxD*

where X are attributes with x being their respective values, and w are a set of weights.

Two different ways of observing a linear graph:

Gráfico

Descripción generada automáticamente

the top-down view with the equi-potential lines represent “the altitude” of the data points in the graph. (2 dimensions A1 and A2 with the output dimension being Y ( = the altitude)).

A linear function also partitions the input space into decision regions (e.g: R1 = {x | y(x) > 0})

There exist certain problems with Linear Classification:

Gráfico

Descripción generada automáticamente

in the figure to the left, we see that in R3, the objects are not C1 nor C2, therefore we could label them C3. However, in the green area they are both C1 and C2, what do we do? Similar idea with the figure to the right.

Linear functions have a version called Discriminant Functions. Each object x belongs to a class k for which yk(x) is maximal. (all objects are part of a class to a certain degree.)

Gráfico

Descripción generada automáticamente

**How are these discriminant values calculated?**

Imagen que contiene Calendario

Descripción generada automáticamente

**(Read from Slide 43/45)**