Machine Learning Algorithm Tasting

Machine learning describes the variety of processes by which we analyze data in order to find patterns and then use those patterns to predict *something*.

Overview

X is the data you have (e.g., emails)
Y is the label you want to predict (e.g., spam or not)
Z is the hidden structure of the data (e.g., features that might help you get to the answer like advertisements, forwards, notification emails, etc)

	discrete	**continuous**
supervised	classification e.g., SVM, Naive Bayes	regression e.g., linear, ridge, lasso
unsupervised	clustering e.g., K-Means, latent dirichlet allocation	dimensionality reduction e.g., PCA, maximally informative dimensions, MDS

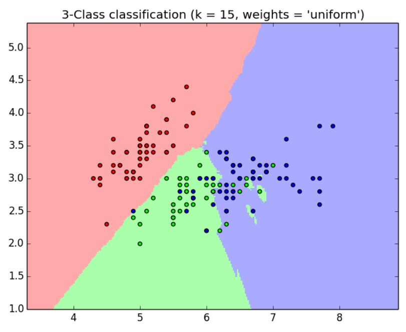
Other: reinforcement learning, ranking, structured prediction

Supervised Methods

In our training data (X,Y) we know what Y is.

K-Nearest Neighbors

What is it
Predicts labels based on how close (or similar) a new point is to existing data. The user determines how many neighbors (K) should be surveyed, and sets a distance metrics (Euclidean most popular, also Mahalanobis). Too small *K* leads to overfitting. Too large *K* just outputs the majority class. It is recommended to use an even value for *K* because it is “too easy” to get an answer from uneven values.



What is it good for
Because it is nonparametric is good for data where the decision boundary is highly irregular.

Neuroimaging Applications
An adaptation of this has been used to measure the similarity of a voxel time series. Authors claim to have increased power and ability to uncover more nuanced activations.

Naive Bayes

What is it
Predicts a class based on the probability of the data. More specifically, it calculates the probability (specify first the class conditional joint probability distribution), assume they are conditionally independent (they are not), then apply Bayes rule to reverse the conditioning.
Classification rule:

$$P(y \mid x_1, \dots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y)$$

$$\Downarrow$$

$$\hat{y} = \arg\max_{y} P(y) \prod_{i=1}^{n} P(x_i \mid y),$$

What is it good for

Neuroimaging Applications

Math…
Flaws/Strengths..
“Naive” because it assumes independence between each pair of features, but this is not usually true. Still, it performs well in practice and it is simple (few parameters) and therefore quite **immune to overfitting**.
Works well with small data sets.
Fast (subverts the *curse of dimensionality*
In neuroimaging

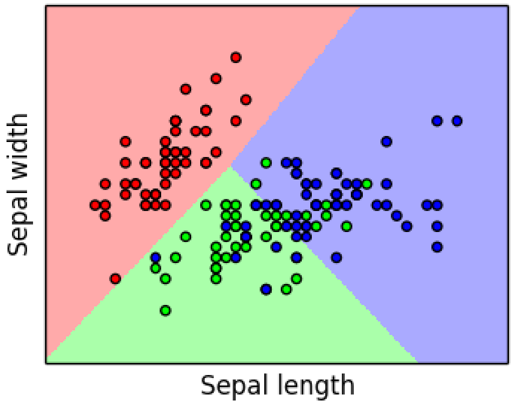
Logistic Regression

What is it
Logistic Regression is a classification method that models the probability of an observation belonging to one of two classes. Results are restricted between 0 & 1 so it has probabilistic interpretations

$$P(Y=0|X)=\sigma(-(\beta_0+\sum_i \beta_i X_i))$$

$$P(Y=1|X)=1-\sigma(-(\beta_0+\sum_i \beta_i X_i))$$

Where $\sigma(z)=\frac{1}{1+exp[-z]}$

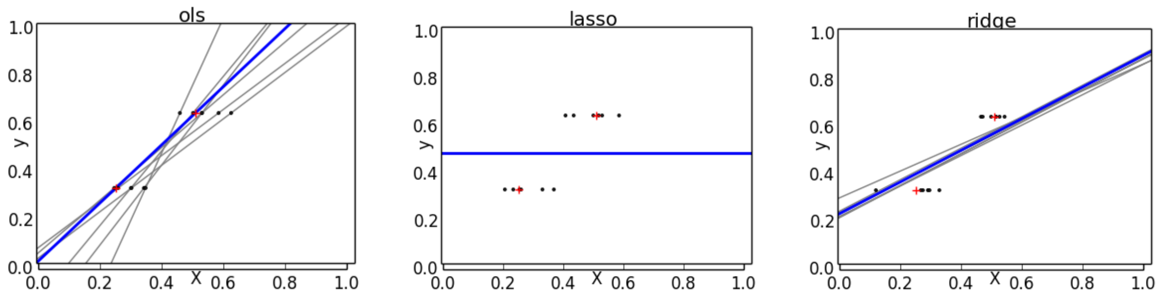


What is it good for

Because it is nonparametric is good for data where the decision boundary is highly irregular.

Neuroimaging Applications

Linear Regression



OLS

LASSO-

The optimization objective for Lasso is:

```
(1 / (2 * n_samples)) * ||y - Xw||^2_2 + alpha * ||w||_1
```

RIDGE

Support Vector Machines

Support vector machines are similar to regression in that they fit a line to separate the data. This line is drawn in way which it maximizes the margin (the distance between the classes). It is the state of art in classification. If the data are not linearly separable, applying a kernel will

increase accuracy.
Aim to draw a line between two classes that is maximally far from any point in the training data.

Equation of hyperplane:
$w \times x_i + b = 0$
Distance of a point to hyperplane:
$$\frac{\left|w \times x_i + b\right|}{||w||}$$
the margin ρis:
$$\rho \equiv min_{(x,y)\in S} \frac{\left|w \times x_i + b\right|}{||w||} = \frac{1}{||w||}$$

What is it good for
Saves memory because you only need to know where the support vectors are (other data does not need to be stored in the classifier).

Neuroimaging Applications

Unsupervised Methods

In our training data (X,Z) we want to discover Z, that is we want to find the hidden structure in data, structure that we can never formally observe.