PREDICTION & CLASSIFICATION

* Labeled data -🡪 predict label

Logistic regression, SVM

* Unlabeled

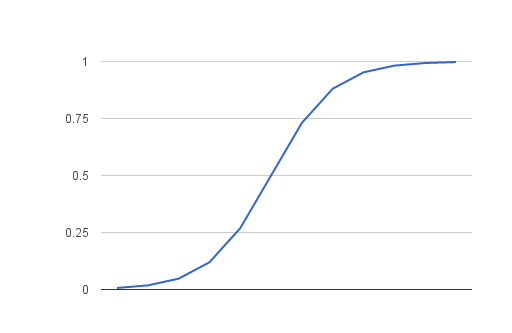
Linear models (polynomial models)

Q: is linear regression prediction or classification?

Y = ax + b

Ei = Yi - (aXi + b)

In logistic regression, we use the sigmoid function



To make a classification (smooth functions)

Assume numeric data set (assume the one we use for our examples)

Q: do we need to re-scale for logistic regression????

Q: what classification methods require scaling?

Any method that uses “distances” must use scaling.

kNN – yes

Naïve Bayesian - NO

Regression

X m, sigma

cX

m2 = c\*m

var\_2 = c\*\*2 \* var

sigma\_2 = c \* sigma\_1

-------------------------------------------

cX + d

mu\_2= c\*mu + d

sigma\_2 = c\*sigma

X1 and X2

X = w1\*X1 + w2\*X2

Example: portfolio = (w1, w2, …, wn)

(w1, w2) ------------🡪 X1 , X2

Let X1 be the distribution of returns for stock1, X2 distribution of returns for stock 2

Y = w1\*X1 + w2\*X2

Q: how to compute m(Y) and sigma(Y)?

M(Y) = w1\*m1 + w2\*m2

X1, x2, …., xn ----------🡪

WX + b --------------🡪

Support Vector machines

Decision Trees

Clustering (k-means)

Random forests, ….

Support Vector machines

HOMEWORK ASSIGNMENT:

1. Required part: use linear SVM to predict your labels

2017 labeled data set (52 labels)

(x, y, {Good, Bad})

X average of 5 daily returns for a week

Y standard dev of these five values

You do not need to re-scale (optional)

Use 2017 to train, then apply your SVM to 2018 and compute accuracy

Optional: take Gaussian SVM and poly SVM, compute accuracy and find out if these are superior to a linear SVM.

Homework on Decision Trees:

1. Take 2017 data, train decision tree classifier and apply to 2018

Compute accuracy (use “entropy” as parameter)

(due in 2 weeks)

kNN, logistic regression, SVM, decision trees:

all of these are “supervised” learning

unsupervised learning:

not given classes, but you want to infer them

what do you need????

Need distance

Given a set of N points

(x1,y1), ………………, (xn, yn)