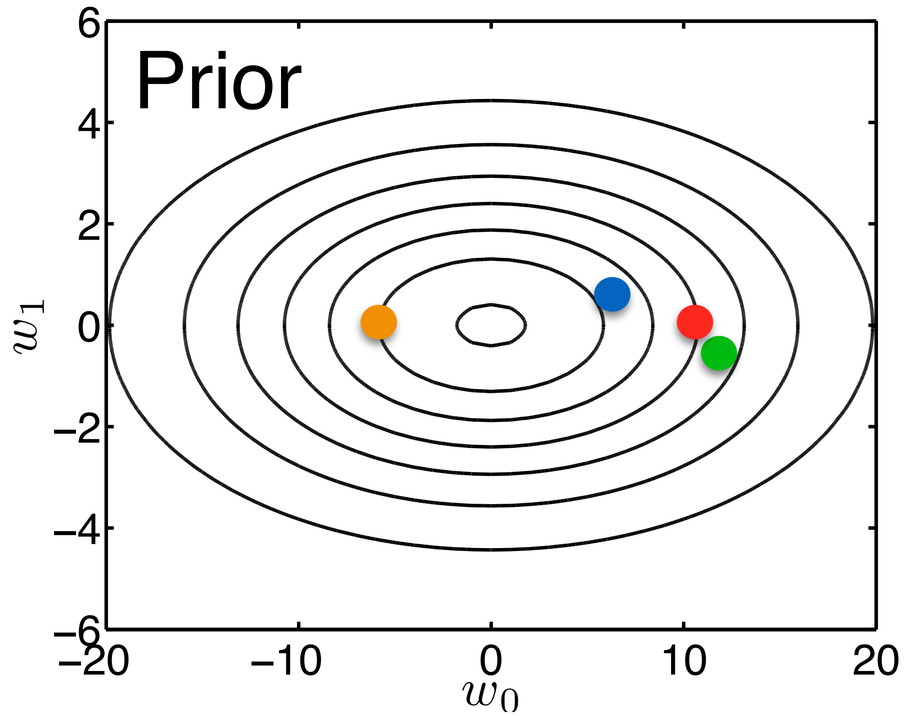
Machine Learning
CS342

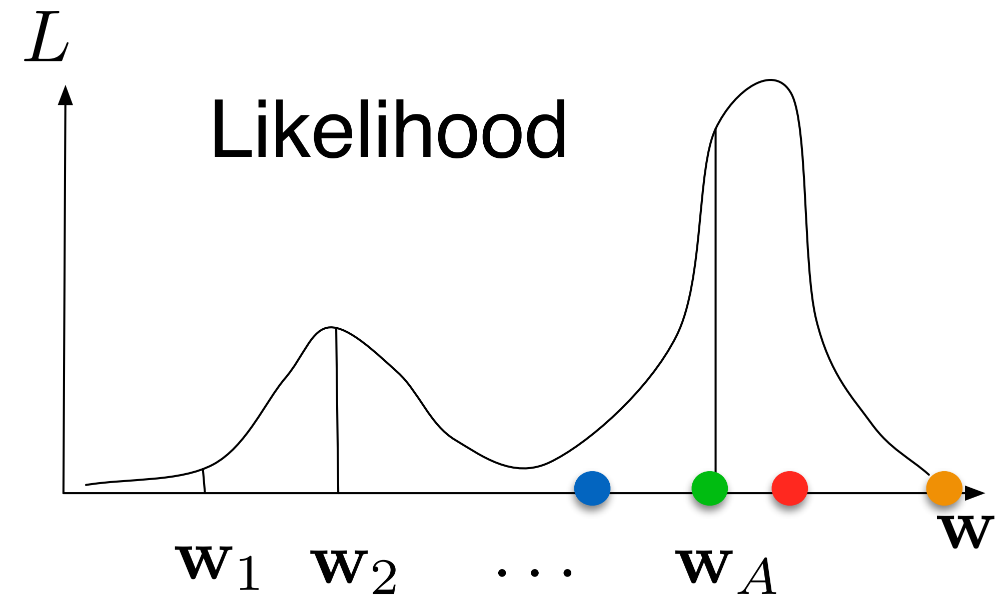
Lecture 11: Probabilistic Classification

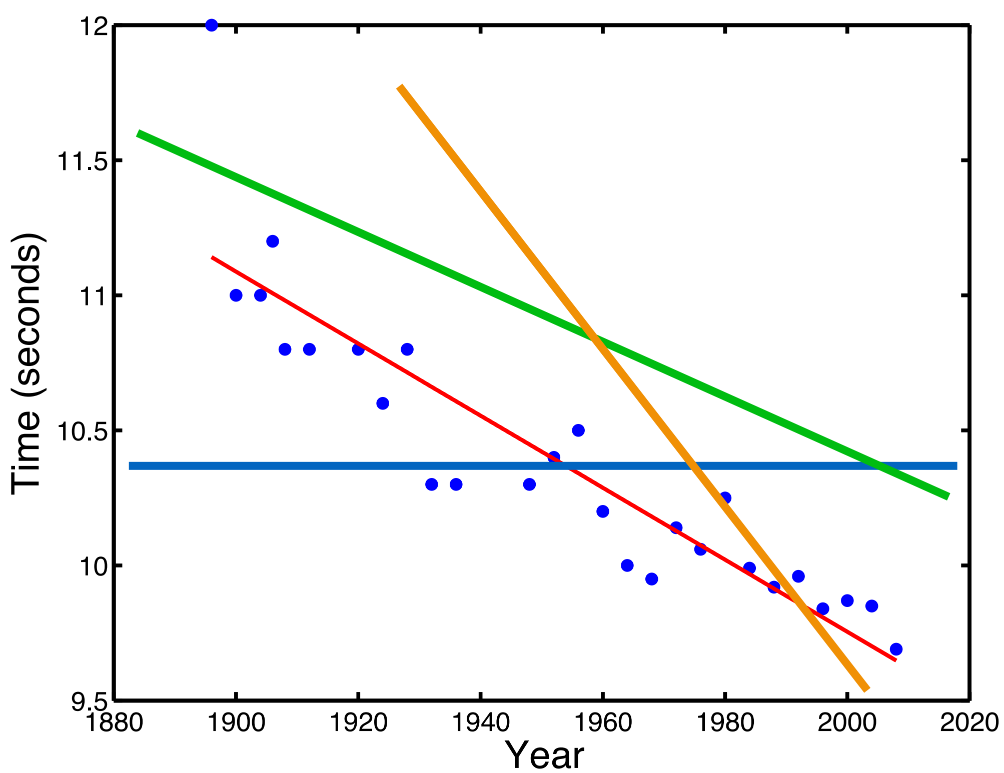
Dr. Theo Damoulas
T.Damoulas@warwick.ac.uk

Office hours: Mon & Fri 10-11am @ CS 307

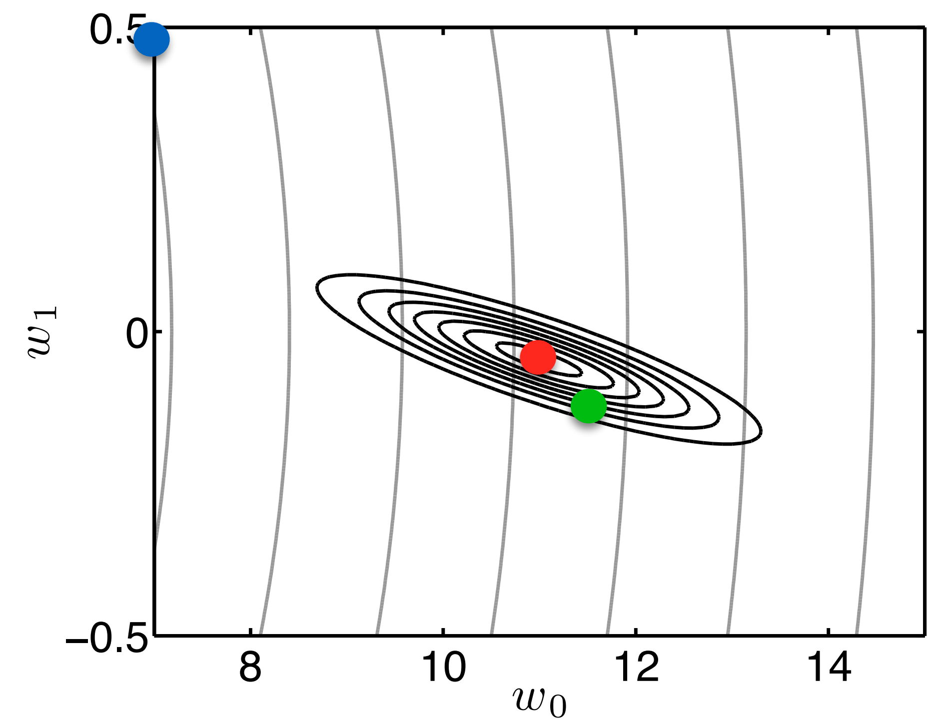
Recap: Probabilistic (Bayesian) Inference







Posterior



Recap: Bayesian Linear Regression

We average over multiple "solutions" **w** to estimate and make predictions

$$\mathbb{E}_{p(\mathbf{w}|\mathbf{X},\mathbf{t})} f(\mathbf{w}) = \int f(\mathbf{w}) p(\mathbf{w}|\mathbf{X},\mathbf{t}) d\mathbf{w}$$

and we see parameters as RVs with associated posterior density:

Bayes Rule

$$\overset{\text{Posterior}}{p(\mathbf{w}|\mathbf{X},\mathbf{t})} = \frac{\overset{\text{Likelihood}}{p(\mathbf{t}|\mathbf{w},\mathbf{X})}\overset{\text{Prior}}{p(\mathbf{w})}}{\underset{\text{Marginal Likelihood}}{p(\mathbf{t}|\mathbf{X})}}$$

- Priors over parameters encode prior knowledge and act as regularisers
- Conjugate priors = closed form posterior densities of same type
- Likelihood = Gaussian, Prior = Gaussian hence Posterior=Gaussian
- Marginal Likelihood is a normalising constant - model selection

Recap: Bayesian Linear Regression

Usually we want a predictive density over t*

predictive density

Gaussian posterior

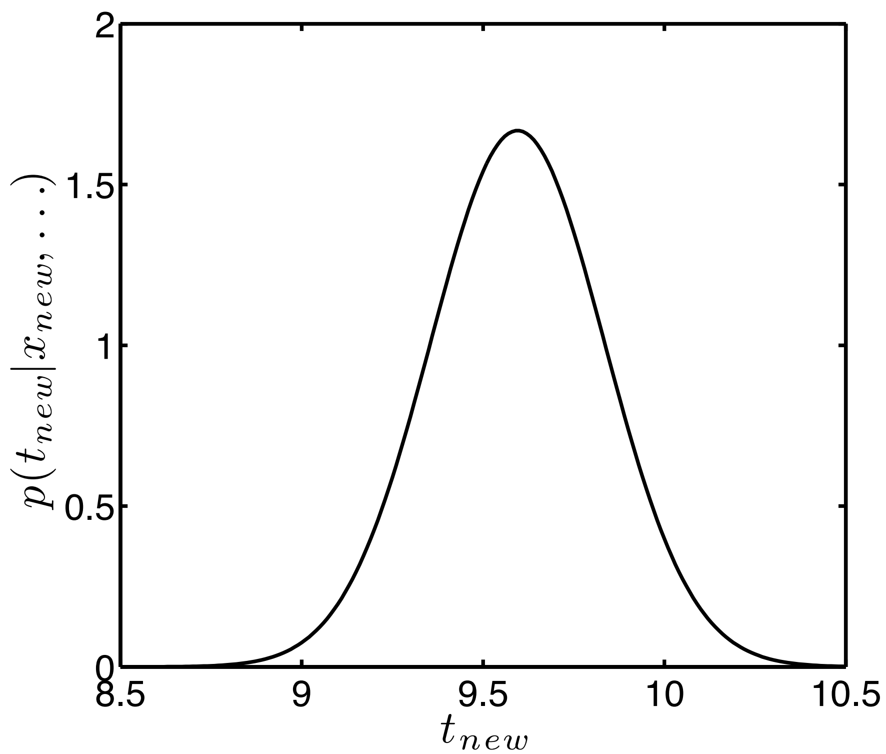
$$p(t^*|\mathbf{x}^*, \mathbf{X}, \mathbf{t}, \sigma^2) = \int p(t^*|\mathbf{x}^*, \mathbf{w}, \sigma^2) p(\mathbf{w}|\mathbf{X}, \mathbf{t}) d\mathbf{w}$$

Gaussian predictive likelihood

So we can output whole densities of probabilities over range of values

predictive density

$$p(t^*|\mathbf{x}^*, \mathbf{X}, \mathbf{t}, \sigma^2)$$



Probabilistic Classification: Bayes classifier

Lets apply our probabilistic framework to classification!

What output do we want at the end from a probabilistic K-class classifier?

$$P(t^* = k | \mathbf{X}, \mathbf{t}, \mathbf{x}^*)$$

Lets use Bayes rule on this directly

$$= \frac{p(\mathbf{x}^* | t^* = k, \mathbf{X}, \mathbf{t}) P(t^* = k)}{\sum_j p(\mathbf{x}^* | t^* = j, \mathbf{X}, \mathbf{t}) P(t^* = j)}$$

Only need to define a Likelihood function and a Prior distribution!

Probabilistic Classification: Bayes classifier

Likelihood...

$p(\mathbf{x}^*|t^* = k, \mathbf{X}, \mathbf{t})$ How likely is **x*** if it is in class k?

We can define any likelihood that makes sense for our data
Think of it as "what sort of density can describe my data from one class?"

For example:

- If our data are D-dimensional vectors of real values:
  Gaussian Likelihood
- If our data are number of heads in N coin tosses:
  Binomial Likelihood

In any case, we use the training data from that class to learn the parameters of that likelihood (e.g. mean, covariance) for every class

Probabilistic Classification: Bayes classifier

Prior…over classes

$$P(t^* = k)$$

Prior probabilities of different classes

Very useful in Imbalanced problems (e.g. medical applications):

e.g. If there are far fewer observations of class 2 then of class 1

$$P(t^* = 1) \gg P(t^* = 2)$$

e.g. If equal

$$P(t^* = 1) = P(t^* = 2) = P(t^* = k) = 1/k$$

Probabilistic Classification: Bayes classifier

We are really done at this point but one variant of this model is well known because of its simplicity

Naive Bayes

Additional independence assumption:
Components/attributes of **x** are independent within each class

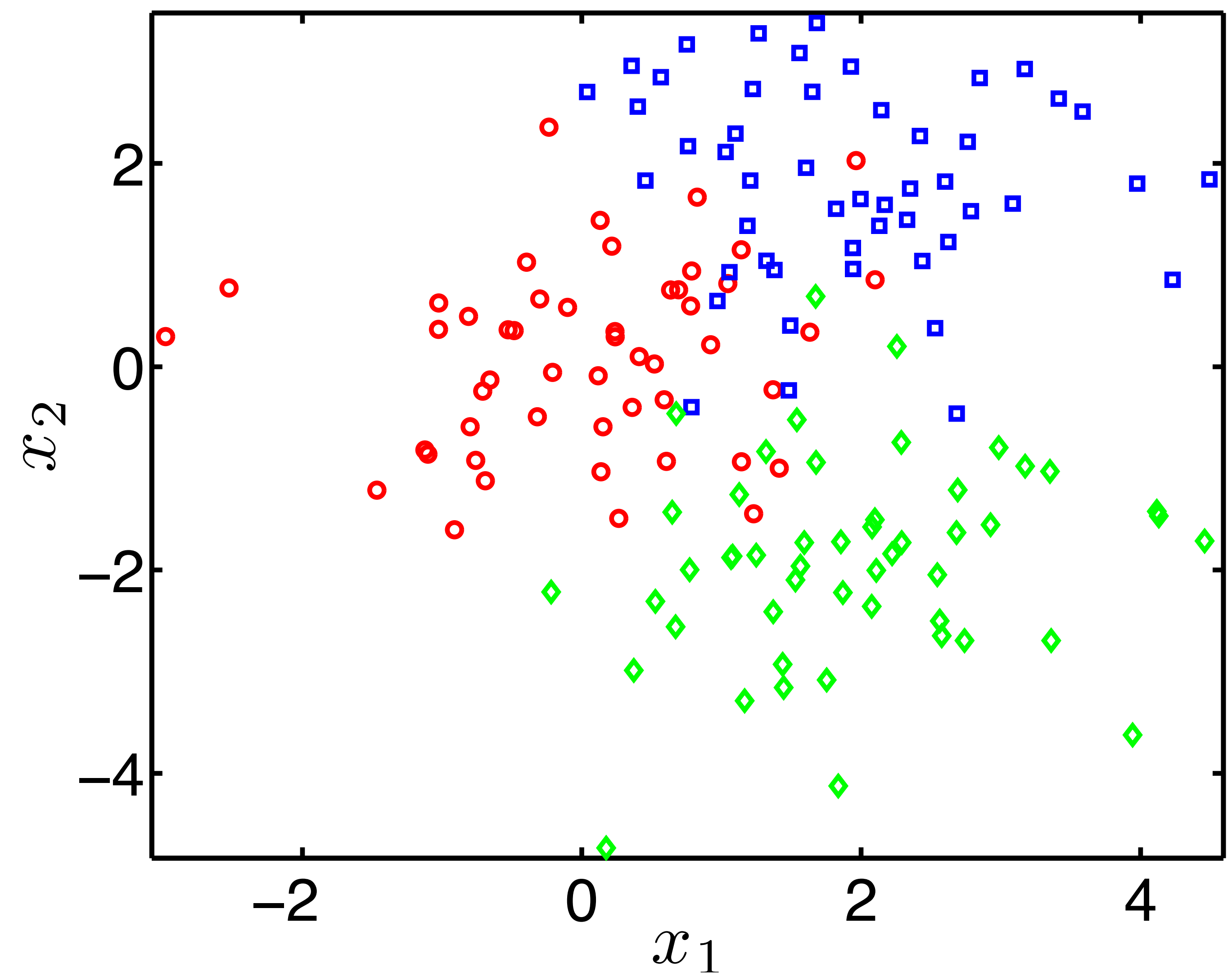
$$p(\mathbf{x}^*|t^* = k, \mathbf{X}, \mathbf{t}) = \prod_{d=1}^{D} p(x_d^*|t^* = k, \mathbf{X}, \mathbf{t})$$

Unrealistic strong assumption - used when we have high-D data

Ok lets combine all that for some real valued data

Bayes classifier example

k=3 classes,
2 attributes x1 and x2,
data real-valued vector



We will use: 1) Gaussian class-conditional distributions (likelihood), 2) the Naive Bayes assumption of independence of attributes per class and, 3) a Uniform prior over class probabilities of 1/k

Probabilistic Classification: Naive Bayes

Again our model:

$$P(t^* = k|\mathbf{X}, \mathbf{t}, \mathbf{x}^*) = \frac{p(\mathbf{x}^*|t^* = k, \mathbf{X}, \mathbf{t})P(t^* = k)}{\sum_j p(\mathbf{x}^*|t^* = j, \mathbf{X}, \mathbf{t})P(t^* = j)}$$

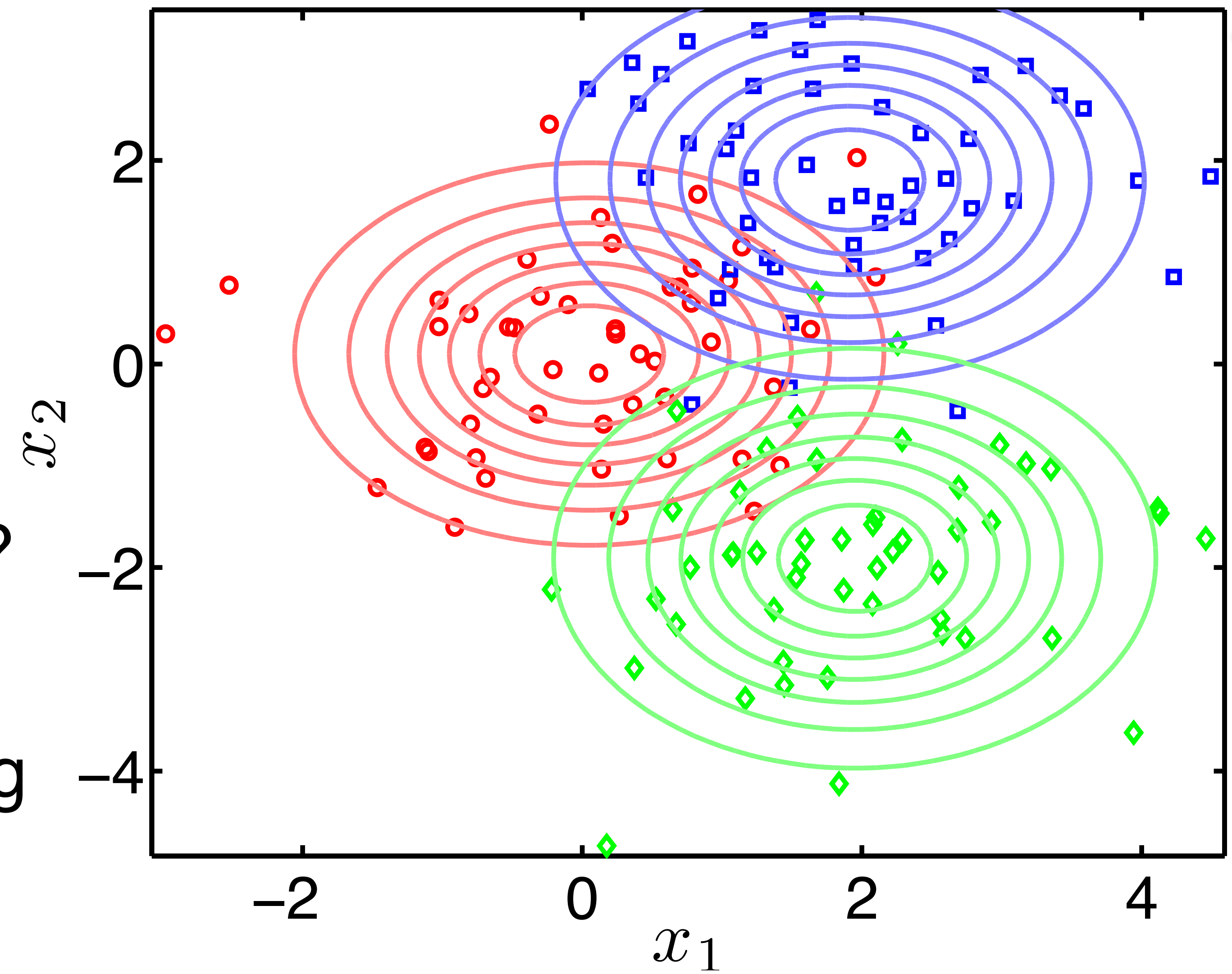
Naive Bayes assumption of independence:
We place a univariate Gaussian on each attribute dimension per class

$$p(\mathbf{x}^*|t^* = k, \mathbf{X}, \mathbf{t}) = \prod_{d=1}^{D} p(x_d^*|t^* = k, \mathbf{X}, \mathbf{t}) = \prod_{d=1}^{D} \mathcal{N}_{x_d^*}(\mu_{kd}, \sigma_{kd}^2)$$

Every dimension of every class is modelled by a univariate normal with its own mean and variance (we will learn these from training data)

Department of COMPUTER SCIENCE

Naive Bayes



Where do you see the NB independence assumption in these Gaussians?

Remember this way of visualising a Gaussian?

Maximum Likelihood estimation!

$$p(\mathbf{x}|t=k,\mathbf{X},\mathbf{t}) = \prod_{d=1}^{2} \mathcal{N}(\mu_{kd},\sigma^2_{kd})$$

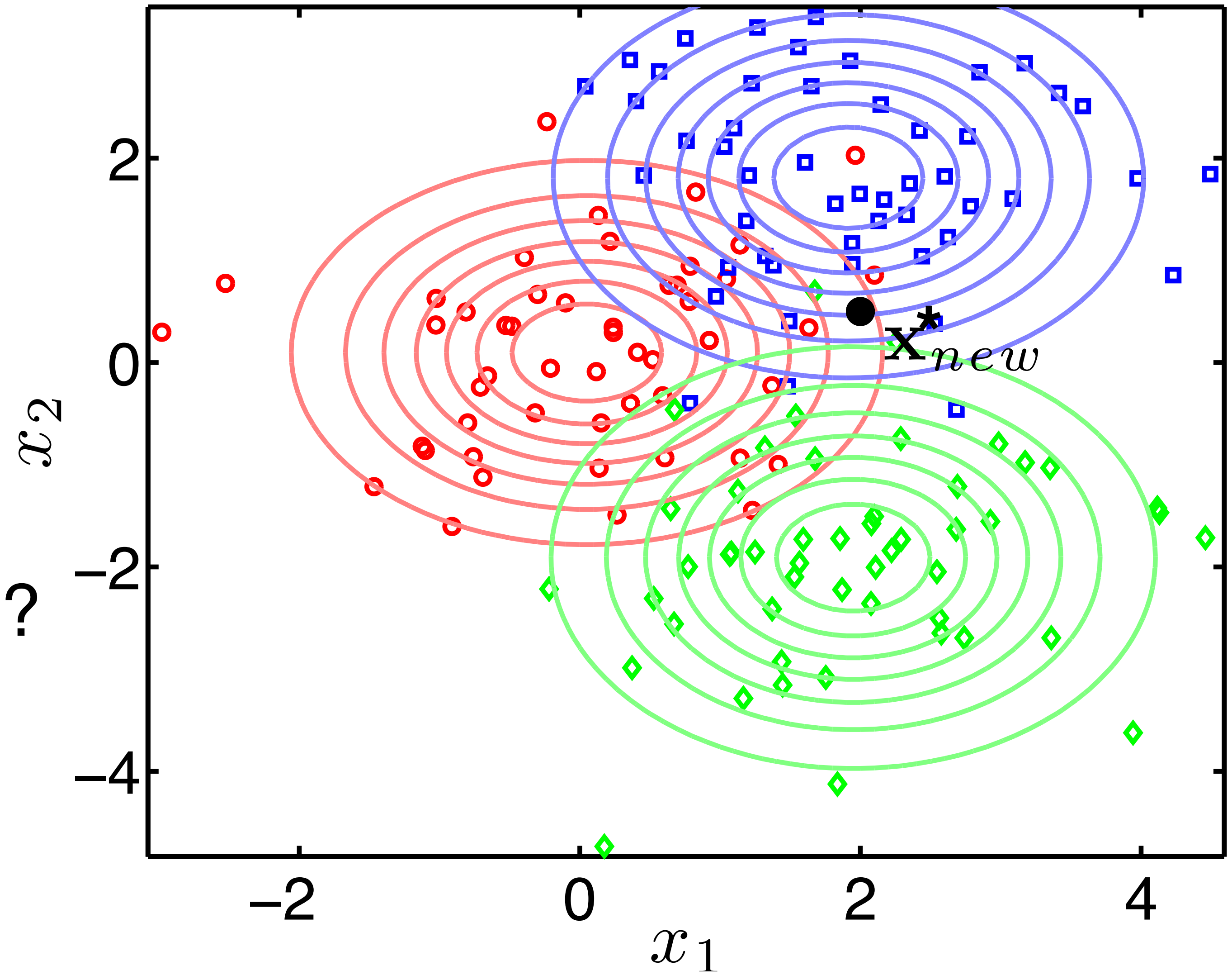
$$\mu_{kd} = \frac{1}{N_k}\sum_{n:t_n=k} x_{nd} \qquad \sigma^2_{kd} = \frac{1}{N_k}\sum_{n:t_n=k}(x_{nd}-\mu_{kd})^2$$

Department of COMPUTER SCIENCE

Naive Bayes

Here is an unseen observation

What class do you think it should be?
How should we decide?



Lets evaluate the densities at the test point **x***

$$p(\mathbf{x}^*|t^* = k, \mathbf{X}, \mathbf{t}) = \prod_{d=1}^{2} \mathcal{N}_{x^*_d}(\mu_{kd}, \sigma^2_{kd})$$

Naive Bayes

Compute Predictions

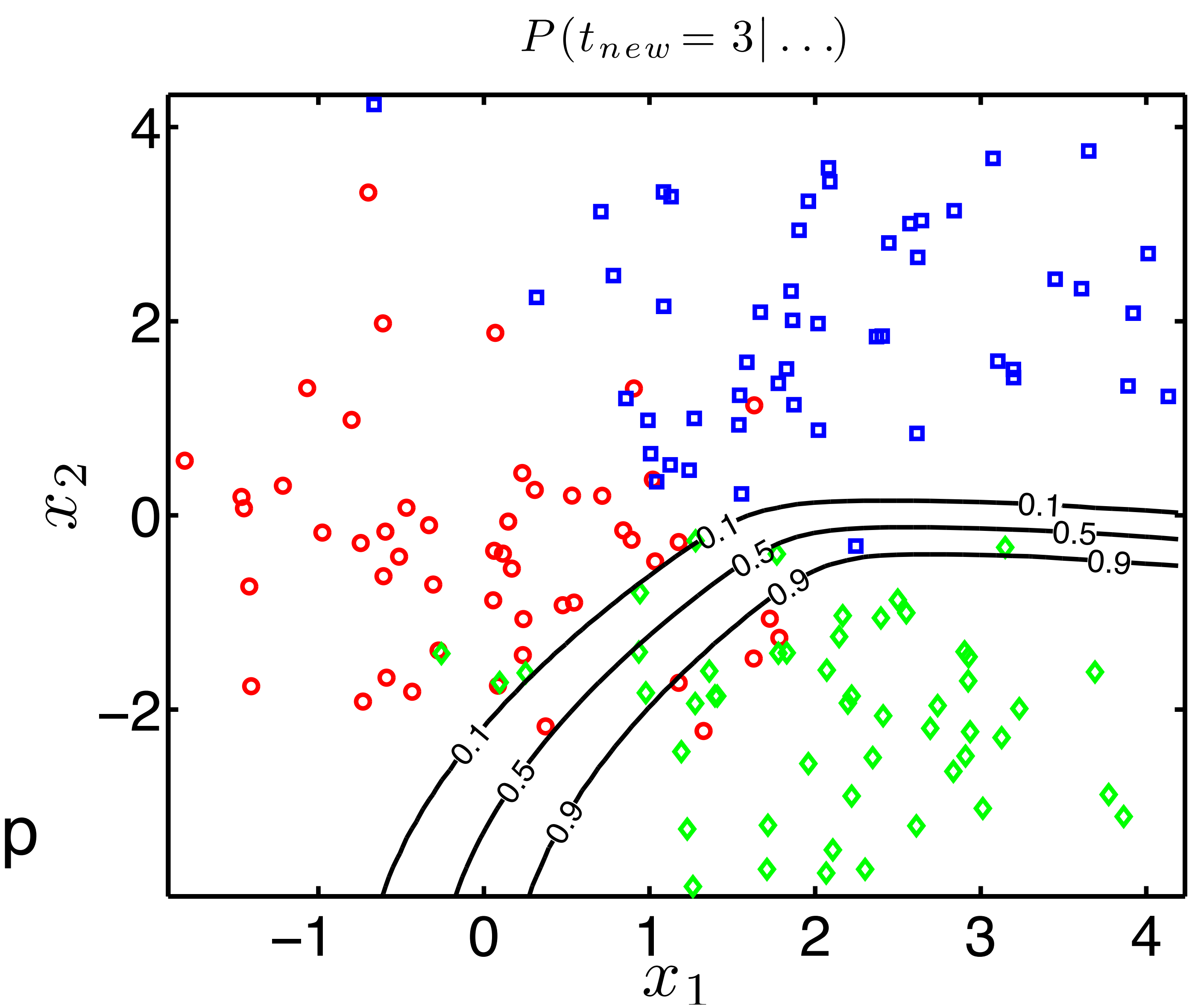
$$P(t^* = k|\mathbf{X}, \mathbf{t}, \mathbf{x}^*) = \frac{p(\mathbf{x}^*|t^* = k, \mathbf{X}, \mathbf{t})P(t^* = k)}{\sum_j p(\mathbf{x}^*|t^* = j, \mathbf{X}, \mathbf{t})P(t^* = j)}$$

Equal class probabilities

Contours of P(t*=k I …)

Non-linear
Decision Boundaries!

You can choose where
to decide for class membership



Bayes classifier (Naive or not)

Used Bayes rule to create a simple probabilistic classifier

- Choose and fit class-conditional densities
  - Maximum Likelihood estimation
  - or we could do fully Bayesian inference by placing priors
- Decide on prior class probabilities
- Compute predictive probabilities
- Naive Bayes variant:
  - Independence assumption of attributes within a class

Obviously you can lift the NB assumption and have a multivariate Gaussian with full covariance matrix

From (NB): $$p(\mathbf{x}|t=k,\mathbf{X},\mathbf{t}) = \prod_{d=1}^{2} \mathcal{N}(\mu_{kd}, \sigma_{kd}^2)$$

To: $$p(\mathbf{x}|t=k,\mathbf{X},\mathbf{t}) = \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma})$$

Generative versus Discriminative models

Modelling the class-conditional densities is a “generative” way of thinking

Generative Framework

A Generative framework is one that tries to model the data generating process. In classification this means that it models the class-conditional densities of the data. You can *generate* new data from the model.

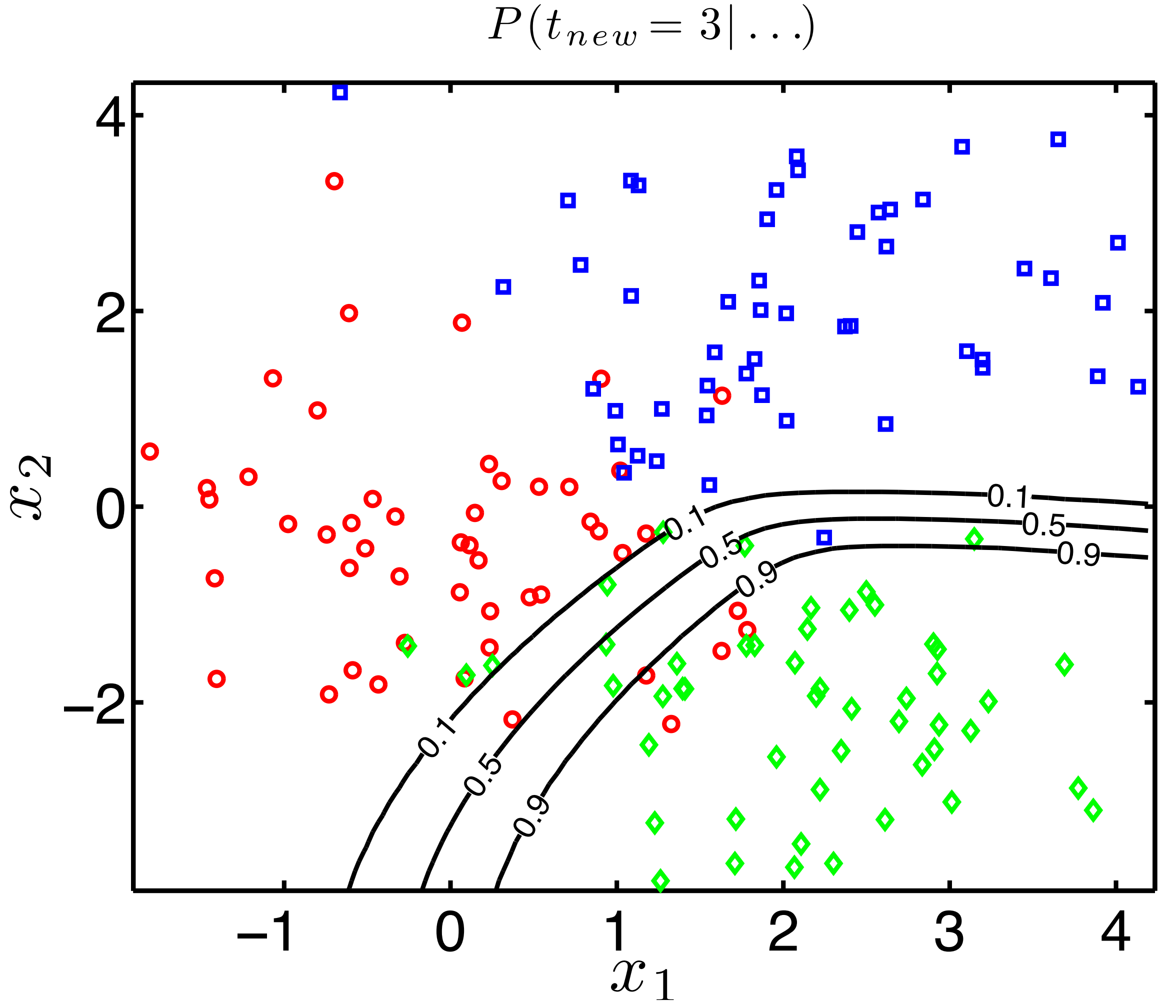
Discriminative Framework

A Discriminative framework is one that tries to model a function that discriminates/separates the classes. This means that it models directly the decision boundary

Some people think all probabilistic (Bayesian) models are “generative” but in fact this is a misconception. We will soon describe a discriminative model within the probabilistic framework

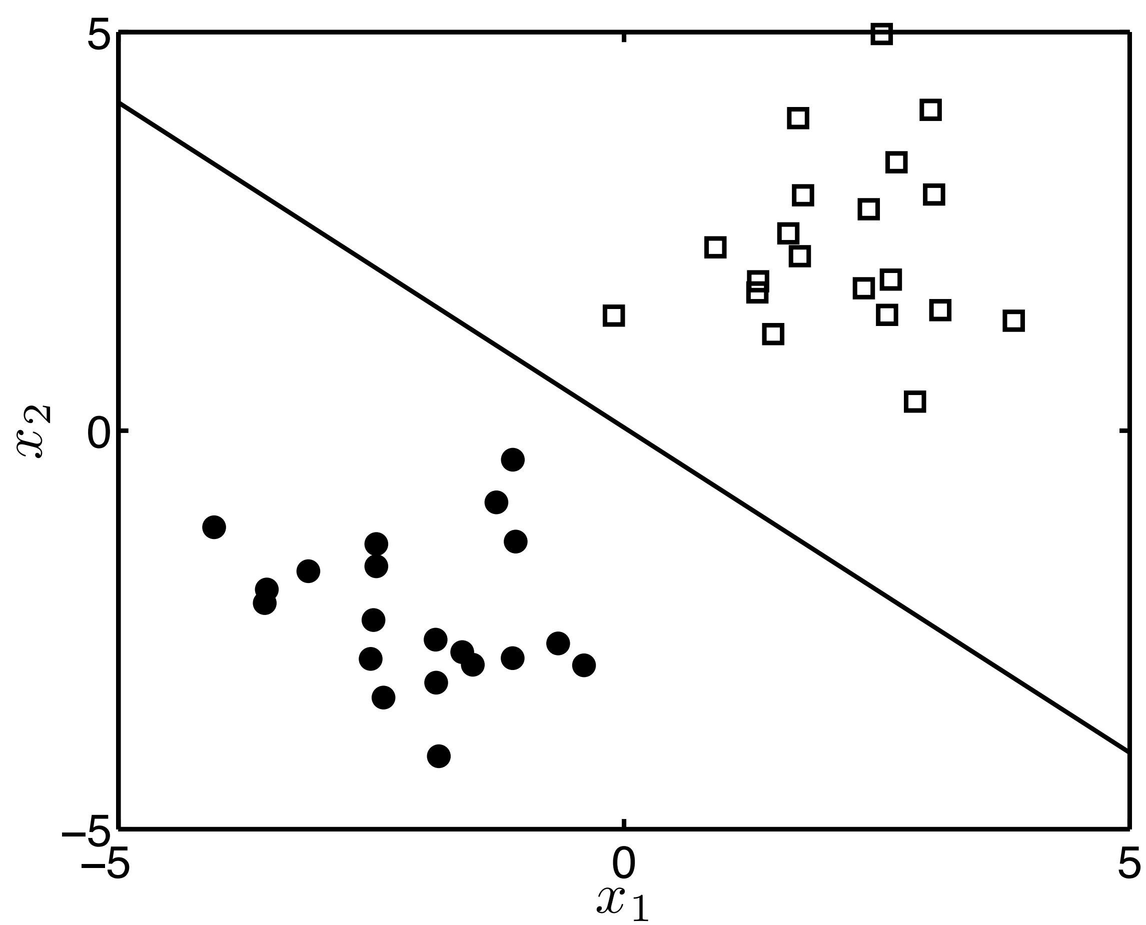
Generative vs Discriminative

Modelling the class-conditional densities (DGP) to get the DB



Generative

Modelling directly the decision boundary (DB)



Discriminative

Generative versus Discriminative models

Generative models, make more assumptions about the DGP as they attempt to model it and that has some dis/advantages:

Think of our 3-class example just now

Advantages: When there isn't much data (really I mean evidence) generative models will probably perform better then a discriminative model due to these strong prior assumptions (more restricted)

Disadvantages: When there is a lot of data (really I mean evidence) generative models might underperform compared to a more flexible discriminative model due to these same assumptions (more restricted)

Generative models address a harder problem (model DGP) then what might be needed (assignment to class)

Logistic regression (Discriminative model)

$$P(t^* = k|\mathbf{X}, \mathbf{t}, \mathbf{x}^*)$$

We applied Bayes rule directly here to get our Generative model:

$$= \frac{p(\mathbf{x}^*|t^* = k, \mathbf{X}, \mathbf{t})P(t^* = k)}{\sum_j p(\mathbf{x}^*|t^* = j, \mathbf{X}, \mathbf{t})P(t^* = j)}$$

Instead lets introduce some parameters and directly model the DB (discriminative model) following a similar path to our Bayesian Linear Regression model

$$P(t^* = k|\mathbf{x}^*, \mathbf{X}, \mathbf{t}) = \int P(t^* = k|\mathbf{x}^*, \mathbf{w})p(\mathbf{w}|\mathbf{X}, \mathbf{t})$$

And we use Bayes rule on the posterior density.. so what do we need?

*Note P() for discrete vs p() for continuous

Logistic regression

Again

$$P(t^* = k|\mathbf{x}^*, \mathbf{X}, \mathbf{t}) = \int P(t^* = k|\mathbf{x}^*, \mathbf{w})p(\mathbf{w}|\mathbf{X}, \mathbf{t})$$

Bayes rule to give posterior density

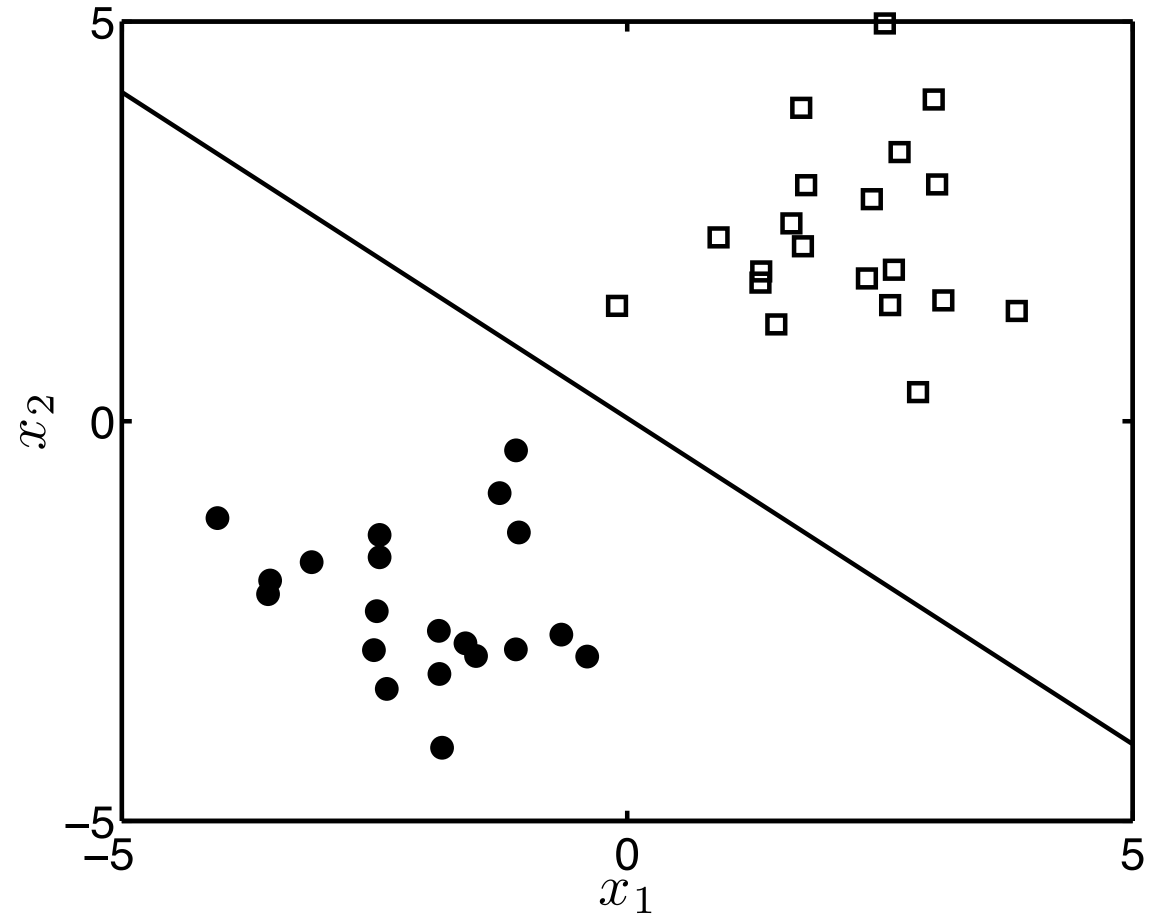
$$p(\mathbf{w}|\mathbf{X}, \mathbf{t}) = \frac{P(\mathbf{t}|\mathbf{w}, \mathbf{X})p(\mathbf{w})}{p(\mathbf{t}|\mathbf{X})}$$

Same thing almost as in Bayesian Linear Regression
What are the quantities we need to define?

Likelihood & Prior distribution(s)

Logistic regression: A new Likelihood function

Focus on Binary Classification - extension to multi-class available



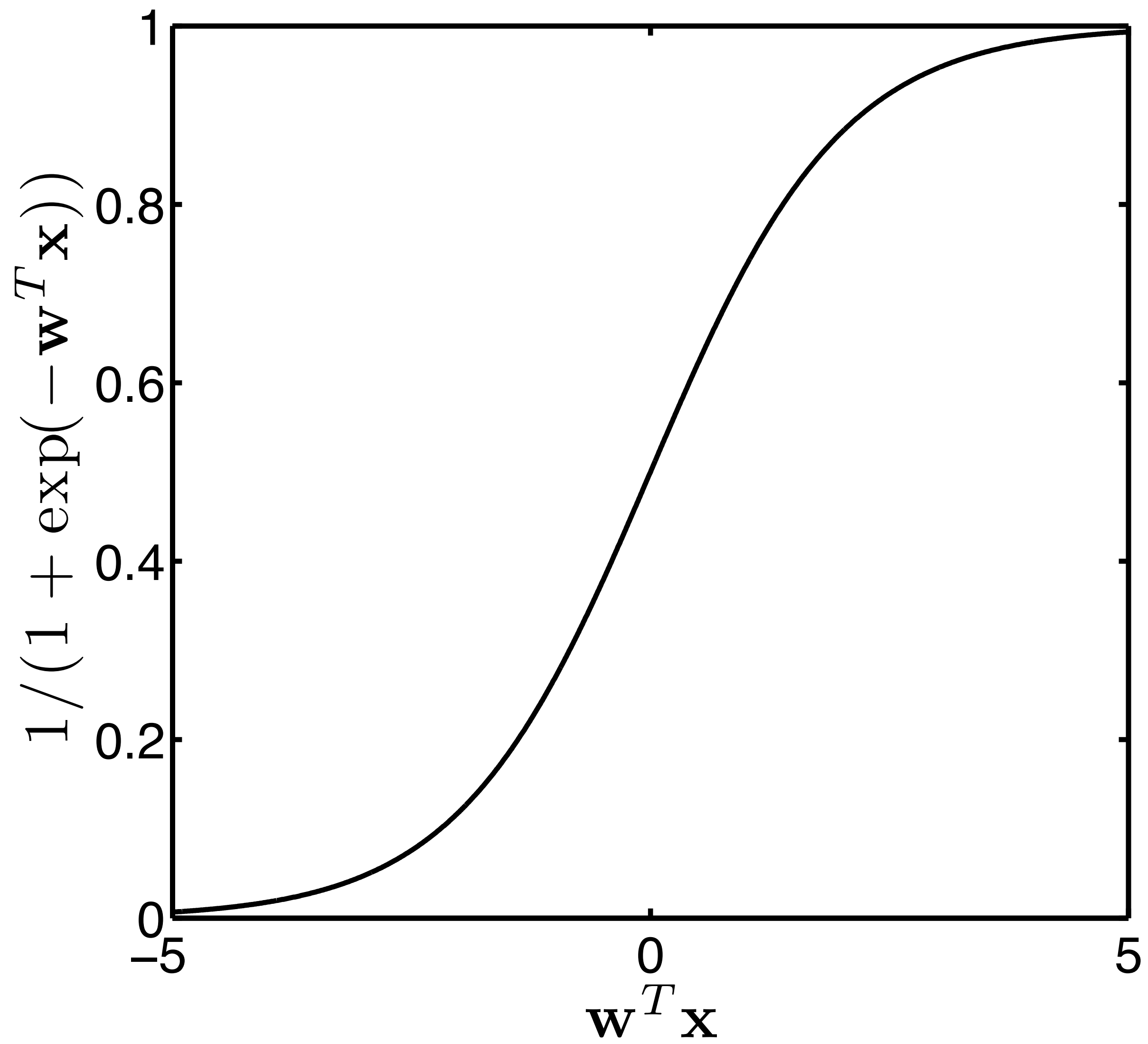
We want a discriminative function e.g. f(**w**)=**xw**=$w_0+w_1x_1+w_2x_2$

Our Likelihood should output a class-membership probability
Need to turn the continuous output of f(**w**) to a probability!

Logistic Likelihood

The Logistic likelihood is a squashing function that does this!
Sigmoid function (looks like an S)

$$P(t = k|\mathbf{w}, \mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{x}\mathbf{w})}$$



Logistic Likelihood: Another motivation for it

It also can been seen as modelling the log-odds with a linear model:

$$\log \frac{P(t=1|\mathbf{w},\mathbf{x})}{P(t=0|\mathbf{w},\mathbf{x})} = \mathbf{x}\mathbf{w} \boxed{= \mathbf{w}^{\mathrm{T}}\mathbf{x} \text{ if } \mathbf{w} \text{ row vector}}$$

Many books use this format

Binary classification so P(t=1 I …)+P(t=0 I …) = 1

Solve for P(t=1 I …) to get:

This is very easy - everyone should be able to derive this!

$$\boxed{P(t=1|\mathbf{w},\mathbf{x}) = \frac{1}{1+\exp(-\mathbf{x}\mathbf{w})}}$$

Logistic likelihood is one of the two most used sigmoid-type likelihoods for classification. The other is the Probit likelihood

To be continued...

We have a Likelihood for binary classificaton with a discriminative model

Logistic Likelihood

$$\boxed{P(t=1|\mathbf{w},\mathbf{x}) = \frac{1}{1+\exp(-\mathbf{x}\mathbf{w})}}$$

We will place a prior density on **w** and attempt to get the posterior density!

Next lecture we will also link this model up back to the OLS/Ridge/Lasso type optimisation farmework!

Some analogies and differences between probabilistic and non-probabilistic models!