TDA231
Classification: Bayes and Naive Bayes

Devdatt Dubhashi
dubhashi@chalmers.se

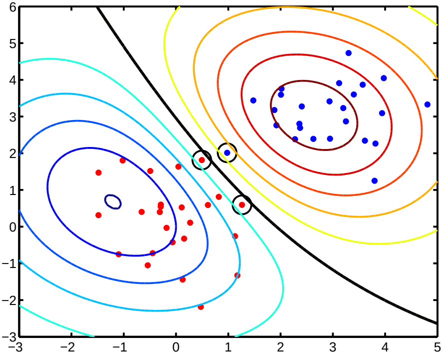
Department of Computer Science and Engg.
Chalmers University

April 9, 2018

Data Representation

- Data objects e.g. email texts or images are represented by fixed dimension vectors, each dimension is called a feature.
- Traditionally (and still) the features were hand–crafted by domain experts (linguists, image researchers).
- Recently with Deep Learning, one tries to learn the features (next week!).

Classification



- A set of N objects with attributes (usually vector) $\mathbf{x}_n$.
- Each object has an associated response (or label) t_n.
- Binary classification: $t_n = \{0, 1\}$ or $t_n = \{-1, 1\}$,
  - (depends on algorithm).
- Multi-class classification: $t_n = \{1, 2, \ldots, K\}$.

Classification

- Input is training data N pairs $(\mathbf{x}_n, t_n), n = 1 \cdots N$.
- Our algorithm should use those to produce a function f that we can apply to a new data point, a test point $\mathbf{x}_{new}$ to classify it.
- Binary classification: $t_n = \{0, 1\}$ or $t_n = \{-1, 1\}$,
  - (depends on algorithm).
- Multi-class classification: $t_n = \{1, 2, \ldots, K\}$.

Classification syllabus

- 4 classification algorithms.
- Of which:
  - 2 are probabilistic.
    - Bayes classifier.
    - Logistic regression.
  - 2 are non-probabilistic.
    - K-nearest neighbours.
    - Support Vector Machines.
- There are many others!

Probabilistic v non-probabilistic classifiers

Classifier is trained on $\mathbf{x}_1, \ldots, \mathbf{x}_N$ and $t_1, \ldots, t_N$ and then used to classify $\mathbf{x}_{\text{new}}$.

- Probabilistic classifiers produce a probability of class membership $P(t_{\text{new}} = k | \mathbf{x}_{\text{new}}, \mathbf{X}, \mathbf{t})$
  - e.g. binary classification: $P(t_{\text{new}} = 1 | \mathbf{x}_{\text{new}}, \mathbf{X}, \mathbf{t})$ and $P(t_{\text{new}} = 0 | \mathbf{x}_{\text{new}}, \mathbf{X}, \mathbf{t})$.
- Non-probabilistic classifiers produce a hard assignment
  - e.g. $t_{\text{new}} = 1$ or $t_{\text{new}} = 0$.
- Which to choose depends on application....

Probabilistic v non-probabilistic classifiers

- Probabilities provide us with more information – $P(t_{\text{new}} = 1) = 0.6$ is more useful than $t_{\text{new}} = 1$.
  - Tells us how **sure** the algorithm is.
- Particularly important where cost of misclassification is high and imbalanced.
  - e.g. Diagnosis: telling a diseased person they are healthy is much worse than telling a healthy person they are diseased.
- Extra information (probability) often comes at a cost.
- For large datasets, might have to go with non-probabilistic.

Bayes classifier

- Our first probabilistic classifier is based on Bayes rule:

$$P(t_{\text{new}} = k|\mathbf{X}, \mathbf{t}, \mathbf{x}_{\text{new}}) = \frac{P(\mathbf{x}_{\text{new}}|t_{\text{new}} = k, \mathbf{X}, \mathbf{t})P(t_{\text{new}} = k)}{\sum_j p(\mathbf{x}_{\text{new}}|t_{\text{new}} = j, \mathbf{X}, \mathbf{t})P(t_{\text{new}} = j)}$$

- We need to define a likelihood and a prior and we're done!

Bayes classifier – likelihood

$$p(\mathbf{x}_{\text{new}} | t_{\text{new}} = k, \mathbf{X}, \mathbf{t})$$

- How likely is $\mathbf{x}_{\text{new}}$ if it is in class k? (not necessarily a probability...)
- We are free to define this *class-conditional distribution* as we like.
- Will depend on type of data.
- e.g.
  - Data are D-dimensional vectors of real values – Gaussian likelihood.
  - Data are number of heads in N coin tosses – Binomial likelihood.
- In both cases, training data with $t = k$ used to determine parameters of likelihood for class k (e.g. Gaussian mean and covariance).

Bayes classifier – prior

$$P(t_{\text{new}} = k)$$

- $\mathbf{x}_{\text{new}}$ not present.
- Used to specify prior probabilities for different classes.
- e.g.
  - There are far fewer instances of class 0 than class 1: $P(t_{\text{new}} = 1) > P(t_{\text{new}} = 0)$.
  - No prior preference: $P(t_{\text{new}} = 0) = P(t_{\text{new}} = 1)$.
  - Class 0 is very rare: $P(t_{\text{new}} = 0) \ll P(t_{\text{new}} = 1)$.

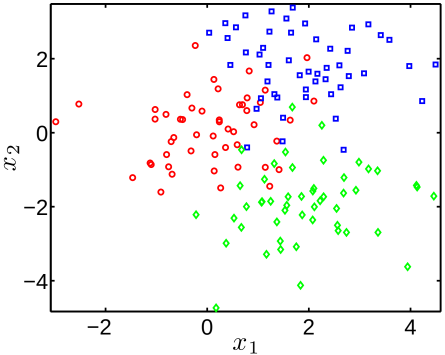
Naive-Bayes

- Naive-Bayes makes the following additional likelihood assumption:
- The components of $\mathbf{x}_{\text{new}}$ are independent for a particular class:

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

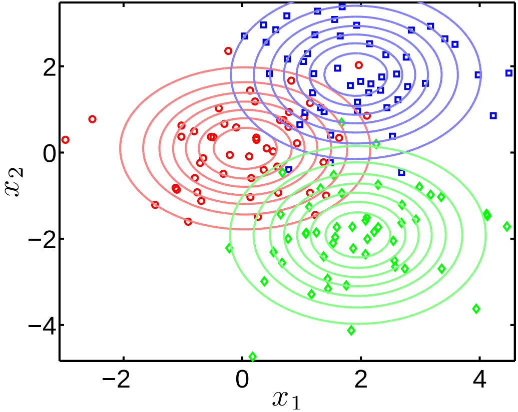
- Where D is the number of dimensions and x_d^{new} is the value of the dth one.
- Often used when D is high:
  - Fitting D uni-variate distributions is easier than fitting one D-dimensional one.

Bayes classifier, example 1



- Each object has two attributes: $\mathbf{x} = [x_1, x_2]^\mathsf{T}$.
- $K = 3$ classes.
- We'll use Gaussian class-conditional distributions (with Naive-Bayes assumption).
- $P(t_{\text{new}} = k) = 1/K$ – uniform prior.

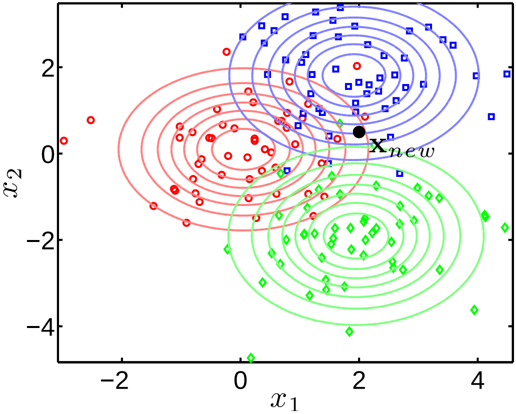
Step 1: fitting the class-conditional densities



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

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

Step 2: Evaluate densities at test point

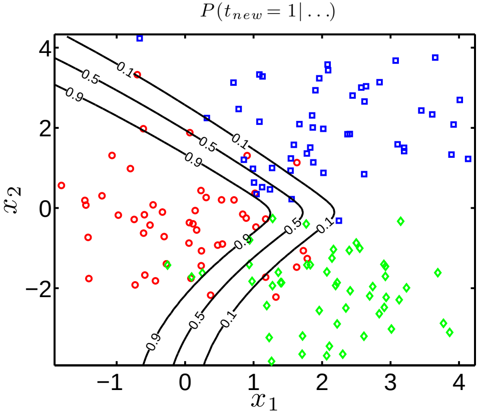


$$p(\mathbf{x}_{\text{new}} | t_{\text{new}} = k, \mathbf{X}, \mathbf{t}) = \prod_{d=1}^{D} \mathcal{N}(\mu_{kd}, \sigma^2_{kd})$$

Compute predictions

- Remember that we assumed $P(t_{\text{new}} = k) = 1/K$.

$$P(t_{\text{new}} = k|\mathbf{x}_{\text{new}}, \mathbf{X}, \mathbf{t}) = \frac{p(\mathbf{x}_{\text{new}}|t_{\text{new}} = k, \mathbf{X}, \mathbf{t})p(t_{\text{new}} = k)}{\sum_j p(\mathbf{x}_{\text{new}}|t_{\text{new}} = j, \mathbf{X}, \mathbf{t})P(t_{\text{new}} = j)}$$

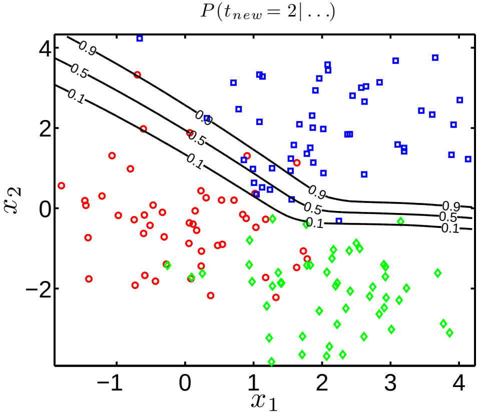


Contours of $P(t_{\text{new}} = 1|\mathbf{x}_{\text{new}}, \mathbf{X}, \mathbf{t})$

Compute predictions

- Remember that we assumed $P(t_{\text{new}} = k) = 1/K$.

$$P(t_{\text{new}} = k|\mathbf{x}_{\text{new}}, \mathbf{X}, \mathbf{t}) = \frac{p(\mathbf{x}_{\text{new}}|t_{\text{new}} = k, \mathbf{X}, \mathbf{t})p(t_{\text{new}} = k)}{\sum_j p(\mathbf{x}_{\text{new}}|t_{\text{new}} = j, \mathbf{X}, \mathbf{t})P(t_{\text{new}} = j)}$$

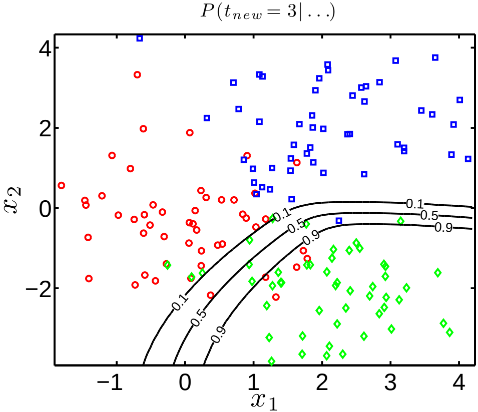


Contours of $P(t_{\text{new}} = 2|\mathbf{x}_{\text{new}}, \mathbf{X}, \mathbf{t})$

Compute predictions

- Remember that we assumed $P(t_{\text{new}} = k) = 1/K$.

$$P(t_{\text{new}} = k|\mathbf{x}_{\text{new}}, \mathbf{X}, \mathbf{t}) = \frac{p(\mathbf{x}_{\text{new}}|t_{\text{new}} = k, \mathbf{X}, \mathbf{t})P(t_{\text{new}} = k)}{\sum_j p(\mathbf{x}_{\text{new}}|t_{\text{new}} = j, \mathbf{X}, \mathbf{t})P(t_{\text{new}} = j)}$$



Contours of $P(t_{\text{new}} = 3|\mathbf{x}_{\text{new}}, \mathbf{X}, \mathbf{t})$

Bayes classifier, example 2

- Data are number of heads in 20 tosses (repeated 50 times for each) from one of two coins:
  - Coin 1 ($t_n = 0$): $x_n = 4,\ 7,\ 7,\ 7,\ 4, \ldots$
  - Coin 2 ($t_n = 1$): $x_n = 18,\ 16,\ 18,\ 14,\ 17, \ldots$
- Use binomial class conditional densities:

$$P(x_n|r_k) = \left(\begin{array}{c} 20 \\ x_n \end{array} \right) r^{x_n}(1-r)^{20-x_n}$$

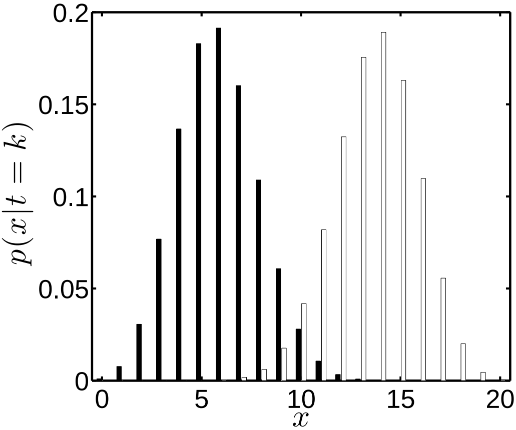
- Where r_k is the probability that coin k lands heads on any particular toss.
- Problem – predict the coin, t_{new} given a new count, x_{new}.
- (Again assume $P(t_{\text{new}} = k) = 1/K$)

Fit the class conditionals...

- Fitting is just finding r_k:

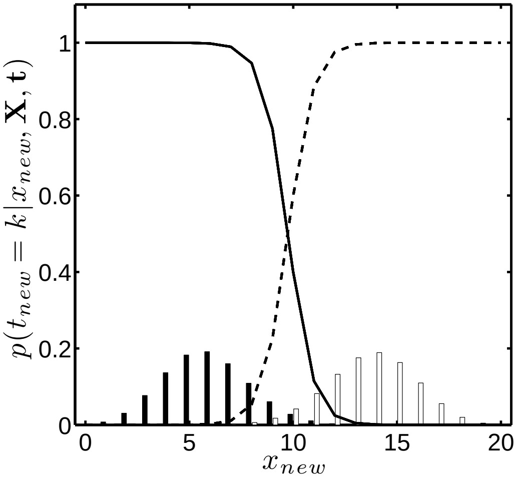
$$r_k = \frac{1}{20N_k} \sum_{n:t_n=k} x_n$$

- $r_0 = 0.287$, $r_1 = 0.706$.



Compute predictions

$$P(t_{\text{new}} = k | x_{\text{new}}, \mathbf{X}, \mathbf{t}) = \frac{p(x_{\text{new}} | t_{\text{new}} = k, \mathbf{X}, \mathbf{t}) P(t_{\text{new}} = k)}{\sum_j p(x_{\text{new}} | t_{\text{new}} = j, \mathbf{X}, \mathbf{t}) P(t_{\text{new}} = j)}$$



Bayes classifier – summary

- Decision rule based on Bayes rule.
- Choose and fit class conditional densities.
- Decide on prior.
- Compute predictive probabilities.
- Naive-Bayes:
  - Assume that the dimensions of $\mathbf{x}$ are independent within a particular class.
  - Our Gaussian used the Naive Bayes assumption (could have written $p(\mathbf{x}|t = k, \ldots)$ as product of two independent Gaussians).