GAUSSIAN NAIVE BAYES

Question?

1) **Assumptions**
2) **Working**
3) **Prediction**

FEW IMP CONCEPTS BEFORE GNB

Generative learning algorithms

GNA belongs to the class of algorithms which comes under GLA.
This type of algorithm try to predict p(x|y) .i.e to learn the type of features associated with each classification, ex when we have output y=0 it tries to find out what is the probability of an output 0 to have a feature vector as x, we can then proceed to find out p(y|x) using simple Bayes rule

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}.$$

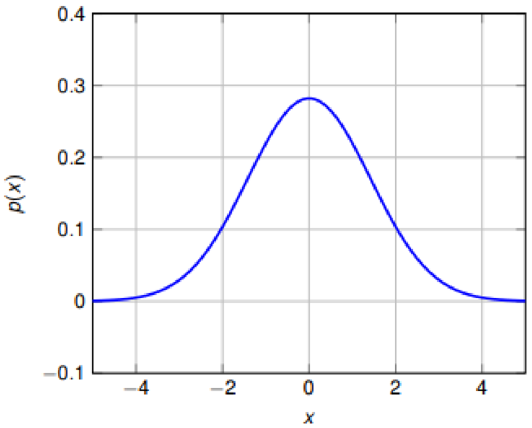
p(x)=p(x|y=1)p(y=1)+p(x|y=0)p(y=0) which is the total probability of output x to i.e being 1 or 0.
p(y) is the total probability of that output.

The normal distribution

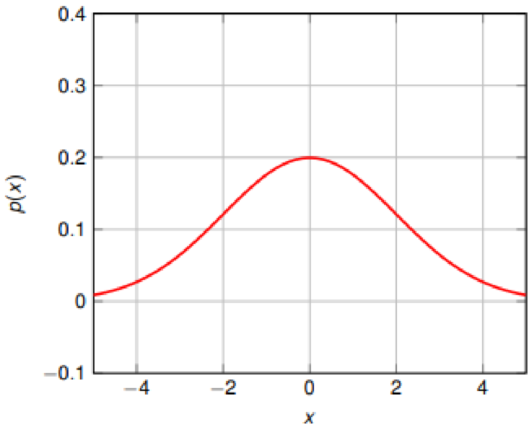
Defining it for one variable

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \quad -\infty < x < \infty,$$

μ is where the function centre will lie.
6 is the variance of the data
And we divide by the term before exp , to give probabilities.



$: \mu = 0, \sigma^2 = 2$



$: \mu = 0, \sigma^2 = 4$

ASSUMPTIONS OF GNA

1)**Normality of Data**: GDA assumes that the features of each class in the dataset follow a Gaussian(normal) distribution.

2) **Independence of features**: Within each class,GDA assumes that data are statistically independent of each other.

GNA is very sensitive to assumptions.

3) **y is distributed as bernoulli.**

Working

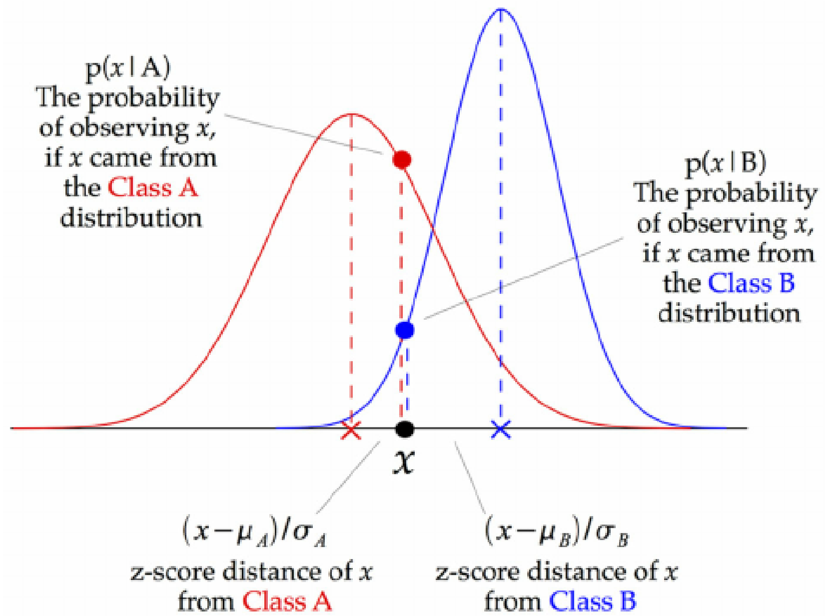
From assumptions of data distribution we have,

$$p(y) = \phi^y(1-\phi)^{1-y}$$

And each feature for each y can be stated as

$$P(X|Y=c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{\frac{-(x-\mu_c)^2}{2\sigma_c^2}}$$

Distribution of each feature looks like,



Now assuming independence of all features we have probability of class as

$$P(y|x_1,\ldots,x_n) = \frac{P(x_1|y)P(x_2|y)\ldots P(x_n|y)P(y)}{P(x_1)P(x_2)\ldots P(x_n)}$$

We are neglecting p(x) in the denominator as it is constant for each class and is needed only to give exact probability.

So

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

Prediction

When we get a data X we calculate likelihood for each possible output by taking product for each feature
And then we take the class with maximum likelihood

$$y = argmax_y P(y) \prod_{i=1}^{n} P(x_i|y)$$

And thus giving us a prediction.