The Bayesian Approach to Machine Learning (Or Anything)

1) We formulate our knowledge about the situation probabilistically:

   – We define a *model* that expresses qualitative aspects of our knowledge (eg, forms of distributions, independence assumptions). The model will have some unknown *parameters*.

   – We specify a *prior* probability distribution for these unknown parameters that expresses our beliefs about which values are more or less likely, before seeing the data.

2) We gather data.

3) We compute the *posterior* probability distribution for the parameters, given the observed data.

4) We use this posterior distribution to:

   – Reach scientific conclusions, properly accounting for uncertainty.

   – Make predictions by averaging over the posterior distribution.

   – Make decisions so as to minimize posterior expected loss.

Finding the Posterior Distribution

The *posterior distribution* for the model parameters given the observed data is found by combining the prior distribution with the likelihood for the parameters given the data.

This is done using *Bayes' Rule*:

$$P(\text{parameters} \,|\, \text{data}) \;=\; \frac{P(\text{parameters})\, P(\text{data} \,|\, \text{parameters})}{P(\text{data})}$$

The denominator is just the required normalizing constant, and can often be filled in at the end, if necessary. So as a proportionality, we can write

$$P(\text{parameters} \,|\, \text{data}) \;\propto\; P(\text{parameters})\, P(\text{data} \,|\, \text{parameters})$$

which can be written schematically as

$$\text{Posterior} \;\propto\; \text{Prior} \times \text{Likelihood}$$

We make predictions by integrating with respect to the posterior:

$$P(\text{new data} \,|\, \text{data}) \;=\; \int_{\text{parameters}} P(\text{new data} \,|\, \text{parameters})\, P(\text{parameters} \,|\, \text{data})$$

Representing the Prior and Posterior Distributions by Samples

The complex distributions we will often use as priors, or obtain as posteriors, may not be easily represented or understood using formulas.

A very general technique is to represent a distribution by a *sample* of many values drawn randomly from it. We can then:

– *Visualize* the distribution by viewing these sample values, or low-dimensional projections of them.

– Make *Monte Carlo* estimates for probabilities or expectations with respect to the distribution, by taking averages over these sample values.

Obtaining a sample from the prior is often easy. Obtaining a sample from the posterior is usually more difficult — but this is nevertheless the dominant approach to Bayesian computation.

A Simple Example — A Hard Linear Classifier

The problem:

We will be observing pairs $(x^{(i)}, y^{(i)})$, for $i = 1, \ldots, n$, where $x = (x_1, x_2)$ is a 2D "input" and y is a $-1/+1$ class indicator. We are interested in predicting y from x. We are not interested in predicting x, and this may not even make sense (eg, we may determine the $x^{(i)}$ ourselves).

Our informal beliefs:

We believe that there is a line somewhere in the input space that determines y perfectly — with -1 on one side, $+1$ on the other.

We think that this line could equally well have any orientation, and that it could equally well be positioned anywhere, as long as it is no more than a distance of three from the origin at its closest point.

We need to translate these informal beliefs into a *model* and a *prior*.

Formalizing the Model

Our model can be formalized by saying that

$$P(y^{(i)} = y \,|\, x^{(i)}, u, w) \;\;=\;\; \begin{cases} 1 & \text{if } y\,u\,(w^T x^{(i)} - 1) > 0 \\ 0 & \text{if } y\,u\,(w^T x^{(i)} - 1) < 0 \end{cases}$$

where $u \in \{-1, +1\}$ and $w = (w_1, w_2)$ are unknown *parameters* of the model. The value of w determines a line separating the classes, and u says which class is on which side. (Here, $w^T x$ is the scalar product of w and x.)

This model is rather dogmatic — eg, it says that y is **certain** to be $+1$ if $u = +1$ and $w^T x$ is greater than 1. A more realistic model would replace the probabilities of 0 and 1 above with ϵ and $1 - \epsilon$ to account for possible unusual items, or for misclassified items. ϵ might be another unknown parameter.

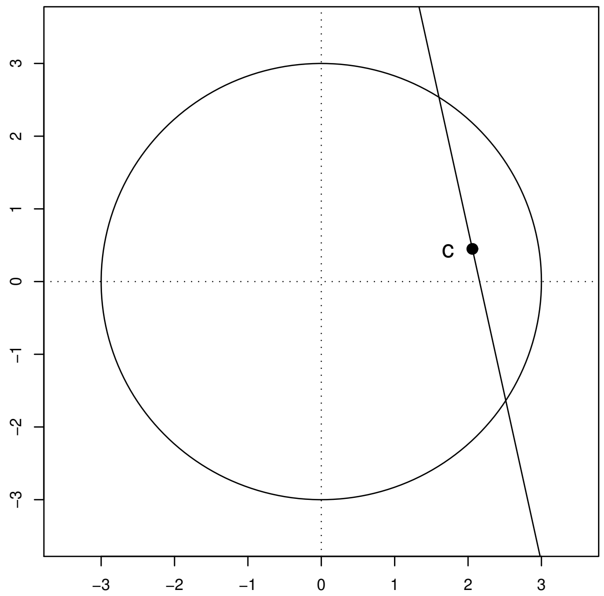
Formalizing the Prior

A line is completely determined by giving the point, c, on the line that is closest to the origin.

To formalize our prior belief that the line separating classes could equally well be anywhere, as long as it is no more than a distance of three from the origin, we decide to use a uniform distribution for c over the circle with radius 3.

Given c, we can compute $w = c/||c||^2$, which makes $w^T x = 1$ for points on the line. (Here, $||c||^2$ is the squared norm, $c_1^2 + c_2^2$.)

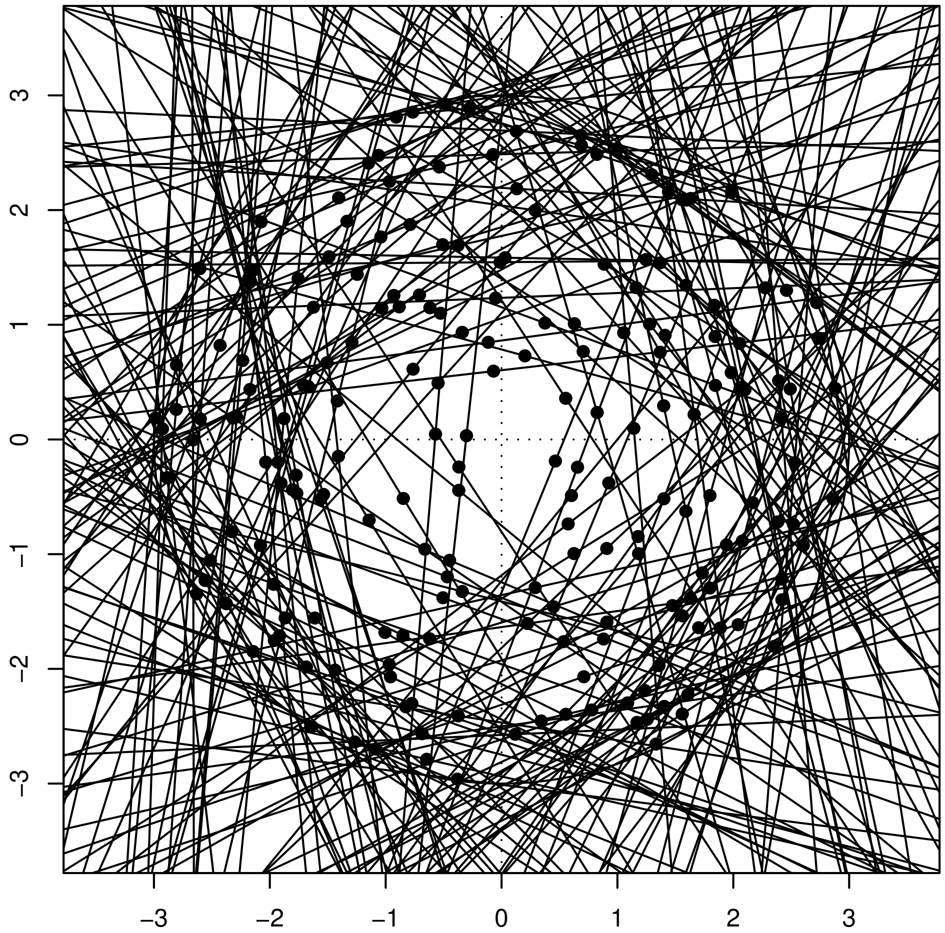
Here's an example:



We also say that u is equally likely to be $+1$ or -1, independently of w.

Looking at the Prior Distribution

We can check this prior distribution by looking at many lines sampled from it:

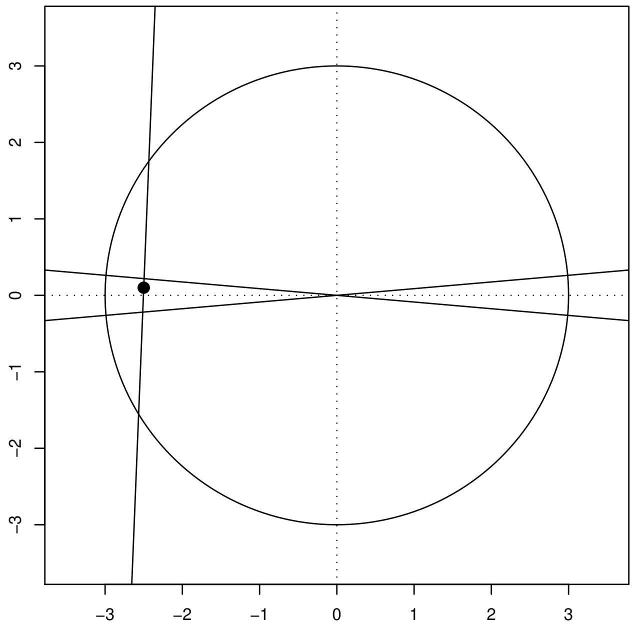


Something's wrong here. We meant for the lines to be uniformly distributed, but we see a sparse region near the origin.

Why This Prior Distribution is Wrong

Our first attempt at formalizing our prior beliefs didn't work. We can see why if we think about it.

Imagine moving a line that's within five degrees of vertical from left to right:

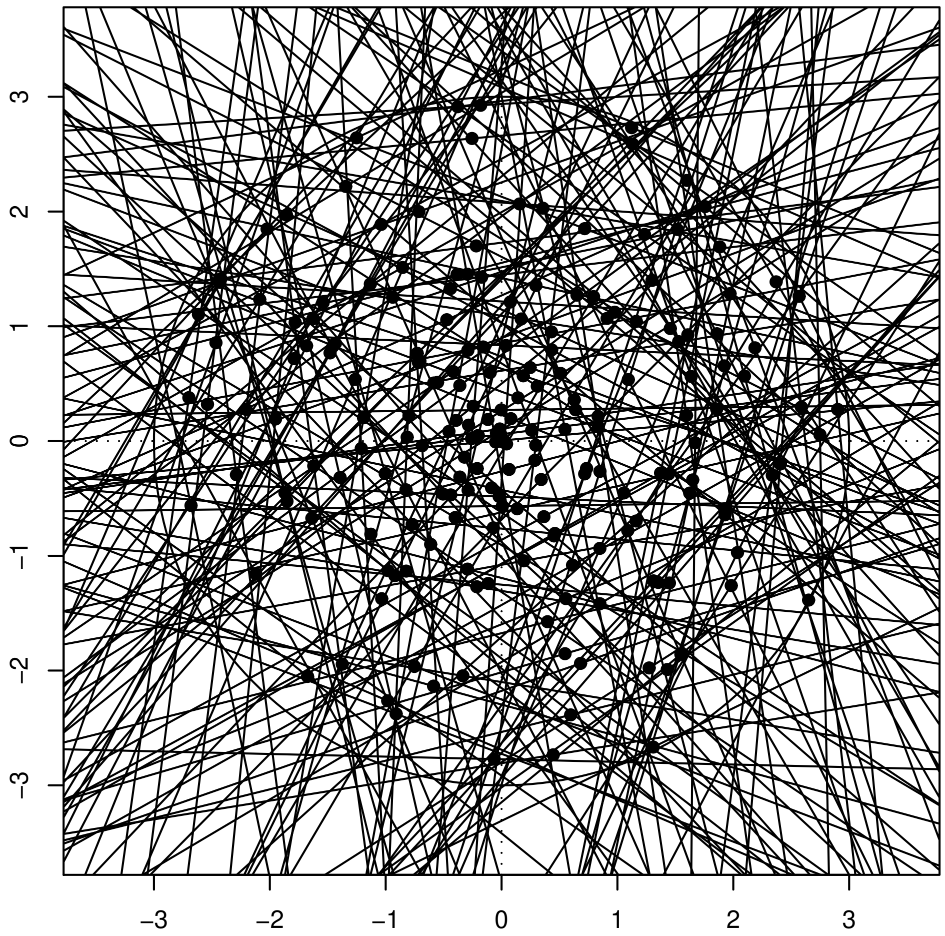


To stay within five degrees of vertical, the closest point to the origin has to be within the wedge shown. This becomes less and less likely as the origin is approached. We don't get the same probability of a near-vertical line for all horizontal positions.

Fixing the Prior Distribution

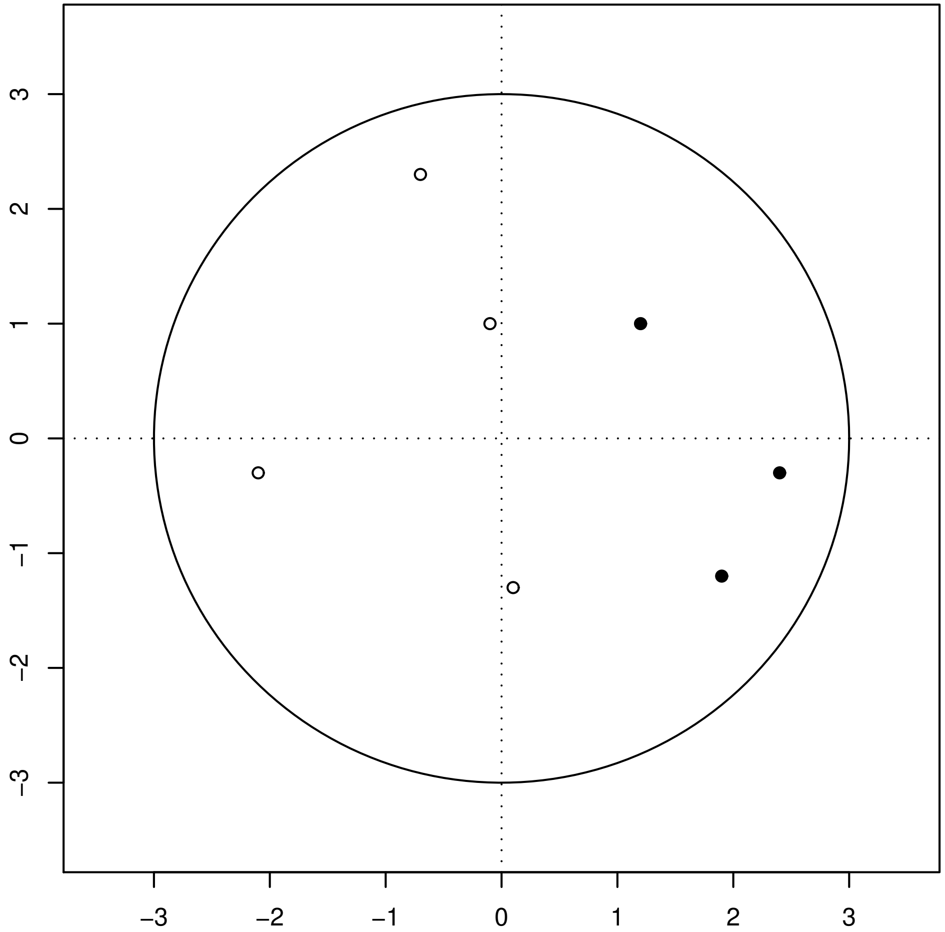
To fix the prior, we lett the closest point on the line to the origin be $c = ru$, with r uniformly distributed over $(0, 3)$ and u uniformly distributed over the unit circle.

Now a sample drawn from the prior looks the way we want it to:



Some Data Points

Now that we have defined our model and prior, let's get some data:



The black points are in class $+1$, the white points in class -1.

Posterior Distribution for the Hard Linear Classifier

For the hard linear classifier, the likelihood is either 0 or 1:

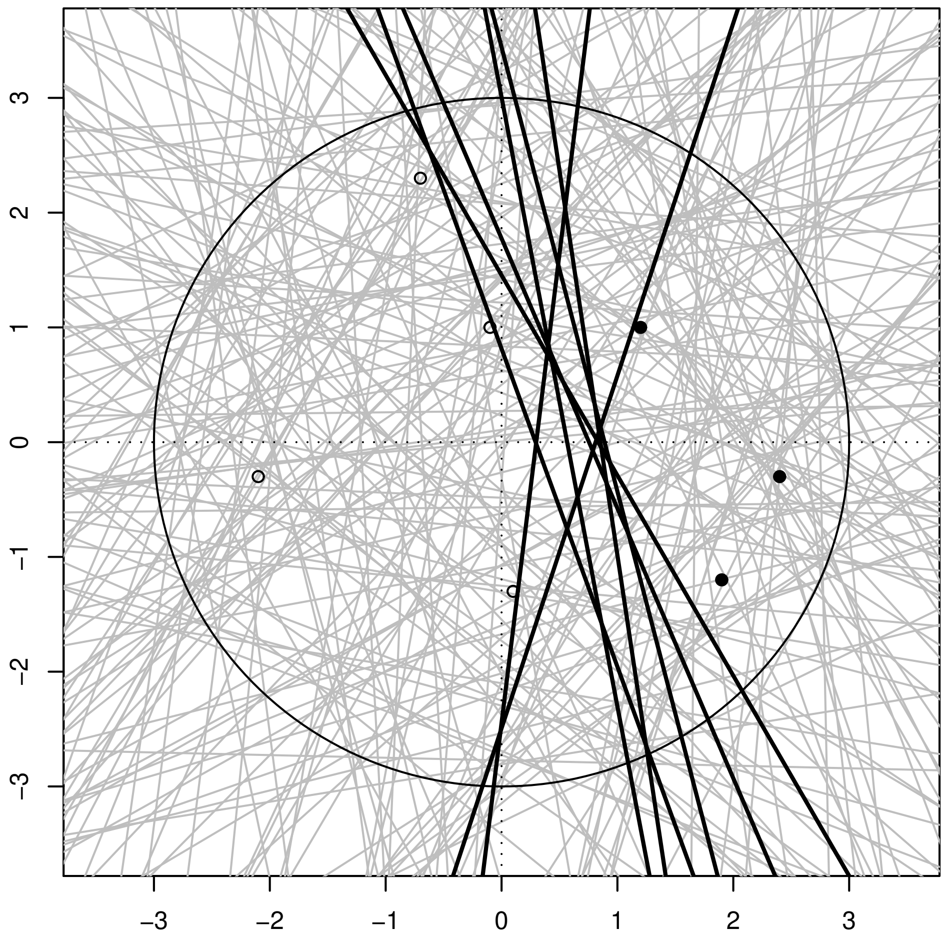
$$
\begin{aligned}
P(y^{(1)},\ldots,y^{(n)} \,|\, x^{(1)},\ldots,x^{(n)},u,w) &= \prod_{i=1}^{n} P(y^{(i)} \,|\, x^{(i)},u,w) \\
&= \begin{cases} 1 & \text{if } y^{(i)}\, u\,(w^T x^{(i)} - 1) > 0, \ \text{ for } i=1,\ldots,n \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
$$

The posterior distribution for u and w is therefore the same as their prior distribution, except that parameter values incompatible with the data are eliminated.

After renormalizing so that posterior probabilities integrate to one, the parameter values compatible with the data will have higher probability than they did in the prior.

Obtaining a Sample from the Posterior Distribution

We sample values from the posterior by sampling w values from the prior, and retaining only those that are compatible with the data (for some u). Example:



The eight bold lines are a random sample from the posterior distribution.

Making a Prediction for a Test Case

The Bayesian predictive probability that in a test case with inputs x^*, the class, y^*, will be $+1$ is found by integrating/summing over the parameters w and u:

$$
\begin{aligned}
&P(y^* = +1 \,|\, x^*, (x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})) \\
&\quad = \int \sum_{u=\pm 1} P(y^* = +1 \,|\, x^*, u, w)\, P(u, w \,|\, x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\, dw
\end{aligned}
$$

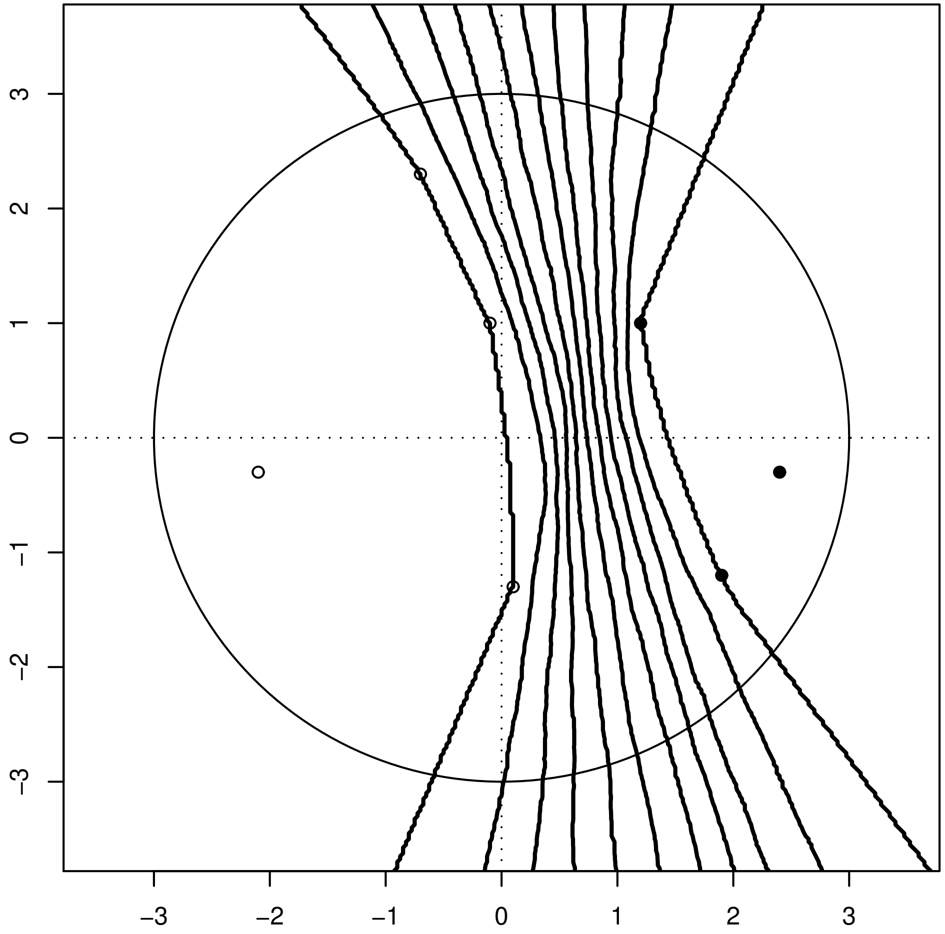
Using a sample of K values from the posterior, $(u^{(1)}, w^{(1)}), \ldots, (u^{(K)}, w^{(K)})$, we can approximate this as follows:

$$
P(y^* = +1 \,|\, x^*, (x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})) \;\approx\; \frac{1}{K} \sum_{j=1}^{K} P(y^* = +1 \,|\, x^*, u^{(j)}, w^{(j)})
$$

For this model, $P(y^* = +1 \,|\, x^*, u^{(j)}, w^{(j)})$ is either 0 or 1, depending on the sign of $u^{(j)}\,({w^{(j)}}^T x^* - 1)$. The average above is just the fraction of lines drawn from the posterior that would put the test point in class $+1$.

A Plot of the Predictive Probabilities

Contour plot over the input space of the predictive probability of class +1 (found using a sample of 10000 parameters from the prior, giving a sample of 450 from the posterior):



The contour lines go from 0 on the left to 1 on the right, in steps of 0.1.

Final Thoughts on This Example

- We see that correctly translating informal knowledge into a prior distribution isn't always trivial.
- However, a prior can be *tested*, by checking how well it corresponds to our prior beliefs. Prior distributions are **not** "arbitrary".
- More elaborate priors might sometimes be appropriate. For example, we might use a prior that favoured lines that are almost horizontal or vertical, if we believe that probably one of the two inputs is mostly irrelevant.
- For a data set with seven points, only about 4.5% of the parameter values we drew from the prior made it into the posterior sample. This technique isn't going to work for realistic problems. We need better ways of sampling from the posterior distribution.

Distinctive Features of the Bayesian Approach

Probability is used not only to describe "physical" randomness (eg, errors in labeling) but also uncertainty regarding the true values of the parameters. Such probabilities represent **degrees of belief**.

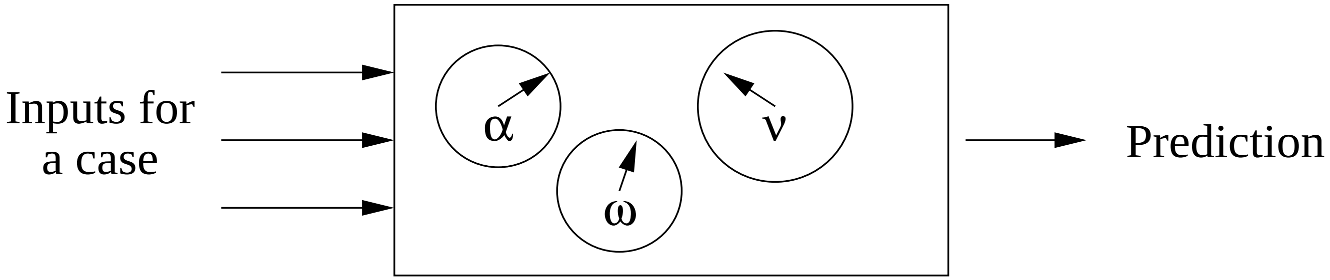
The Bayesian approach takes **modeling** seriously. A Bayesian model includes a suitable prior distribution for model parameters. If the model/prior are chosen without regard for the actual situation, there is *no justification* for believing the results of Bayesian inference.

The model and prior are chosen based on our knowledge of the problem. These choices are **not**, in theory, affected by the amount of data collected, or by the question we are interested in answering. We **do not**, for example, restrict the complexity of the model just because we have only a small amount of data.

Pragmatic compromises are inevitable in practice — eg, no model and prior perfectly express to our knowledge of the situation. The Bayesian approach relies on reducing such flaws to a level where we think they won't seriously affect the results. If this isn't possible, it may be better to use some other approach.

Contrast With the "Learning Machine" Approach

One view of machine learning pictures a "learning machine", which takes in inputs for a training/test case at one end, and outputs a prediction at the other:



The machine has various "knobs", whose settings change how a prediction is made from the inputs. Learning is seen as a procedure for twiddling the knobs in the hopes of making better predictions on test cases — for instance, we might use the knob settings that minimize prediction error on training cases.

This approach differs profoundly from the Bayesian view:

– The choice of learning machine is essentially *arbitrary* — unlike a model, the machine has no meaningful semantics, that we could compare with our beliefs.

– The "knobs" on the machine *do not* correspond to the parameters of a Bayesian model — Bayesian predictions, found by averaging, usually cannot be reproduced using *any* single value of the model parameters.