An introduction to Bayesian probabilities and their application

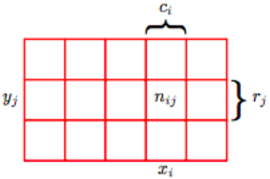
Kieran Gorman

July 24, 2015

Probability

- How 'likely' is the occurrence of a certain event?
- Traditionally 'frequentist'.
- The limit given relative observed frequency.
- That is, having observed $P(x) \approx \frac{n_x}{n_t}$ we might deduce that $P(x) = \lim_{x \to \infty} \frac{n_x}{n_t}$
- In the frequentist model, a hypothesis is a well defined proposition (i.e. certainly true or certainly false).

Probability cont.



- Joint: $P(X = x_i, Y = y_j) = \frac{n_{ij}}{N}$
- Sum rule: $P(X = x_i) = \sum_{j=1}^{L} P(X = x_i, Y = y_j)$
- Conditional: $P(Y = y_j | X = x_i) = \frac{n_{ij}}{c_i}$
  $P(X = x_i, Y = y_j) = \frac{n_{ij}}{N} = \frac{n_{ij}}{c_i} \cdot \frac{c_i}{N}$
- Product rule: $P(X = x_i, Y = y_j) = P(Y = y_j | X = x_i) \cdot P(X = x_i)$

This also holds for continuous variables, as probability densities, just replace $\sum$ with $\int$

Bayes Theorem

It's just an application of the sum and product rules.

$$
\begin{aligned}
P(X, Y) &= P(Y, X) \\
P(Y|X) \cdot P(X) &= P(X|Y) \cdot P(Y) \\
P(Y|X) &= \frac{P(X|Y) \cdot P(Y)}{P(X)}
\end{aligned}
$$

We're not quite at actual Bayesian probabilities, but we can immediately begin to apply Bayes theorem in a classification problem.

Naïve Bayes Classifier

Apply Bayes theorem, along with a strong independence assumption (this is what makes it 'naïve').

$$\begin{aligned} p(C_k, x_1, \dots, x_n) &= p(C_k)p(x_1, \dots, x_n | C_k) \\ &= p(C_k)p(x_1|C_k)p(x_2, \dots, x_n|C_k, x_1) \\ &= p(C_k)p(x_1|C_k)p(x_2|C_k, x_1)p(x_3, \dots, c_n|C_k, x_1, x_2) \\ &= p(C_k)p(x_1|C_k)p(x_2|C_k, x_1) \dots p(x_n|C_k, x_1, \dots, x_{n-1}) \end{aligned}$$

Now assume all of the values in the input vector are independent of one another, meaning $P(A|B, C) = P(A|B) \cdot P(A|C)$. In our case it means that $p(x_i|C_k, x_j, x_k) = p(x_i|C_k)$.

Naïve Bayes Classifier cont.

So, applying Bayes theorem we get

$$\begin{aligned} P(C_k|x_1,\ldots,x_n) &\propto p(C_k)p(x_1,\ldots,x_n|C_k) \\ &\propto p(C_k)p(x_1|C_k)p(x_2|C_k)\ldots p(x_n|C_k) \\ &\propto p(C_k)\prod_{i=1}^{n} P(x_i|C_k) \end{aligned}$$

That is, the probability of labelling an input instance with a certain class is simply the probability of that class, multiplied by the likelihood that instance appearing in that class based on what we've seen already.

This assumption is clearly false in almost all interesting cases, however in general corpus is king. See "Unreasonable effectiveness of data", from Norvig.

Contrived demonstration

Demo.

A Bayesian approach

We can interpret Bayes theorem as:

$$
\begin{aligned}
p(A|B) &= \frac{p(B|A) \cdot p(A)}{p(B)} \\
&\propto p(B|A) \cdot p(A) \\
\textit{posterior} &\propto \textit{likelihood} \cdot \textit{prior}
\end{aligned}
$$

We can see that by counting frequencies, we are fixing our probability estimator to one that maximises likelihood for computing $p(x)$.
The Bayesian approach is to permit an uncertainty so our likelihood is not beholden to the evidence alone.

A Bayesian approach cont.

- Consider for example events we have a degree of belief in occurring, but we can't readily measure (say, the state of the polar ice caps at the turn of the next century).
- Frequencies require repeatability!
- Bayesian probabilities are sometimes called "a logical approach to probabilities" and can be derived from Cox's axioms (i.e. "degrees of belief") for example.

- In Frequentism we have a fixed input, and hypothesise about possible underlying datasets from which it might derive.
- In Bayesianism, we only have the observed data, and the uncertainty is in a probability distribution over the inputs.
- Using a maximum likelihood function over fixed inputs without catering for uncertainty in the hypothesis can lead to extreme conclusions.

Demo

A slightly more involved demo.

Onwards

- What if we want to model dependencies?
- Need more involved models!
- Bayesian networks: probabilistic graphical models that allow for belief propagation. (Look up Judea Pearl.)
- Can't generally be computed in a closed form.

Iterative models

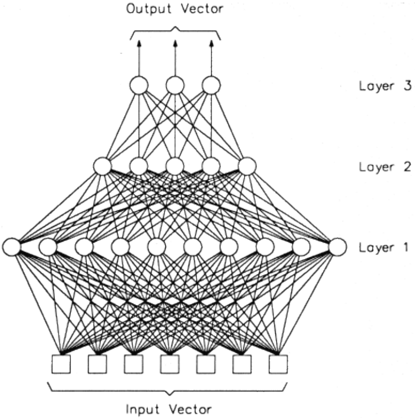
- Because we were using Gaussians as the underlying distribution of the naïve classifier, we could compute the max-likelihood parameter set in a closed form.
- More complex models require a more explicit parameterisation.
- One example: neural networks.

Neural networks from 30,000ft

- Given an input vector, produce an output vector.
- Compare output vector to target with some notion of “fitness”
- Update weightings to increase fitness. Iterate.

Neural networks from 29,999ft

- How to update?
- Back-propagation algorithm.



Gradient descent search

- In general: gradient descent search.
- Exploring a solution space.
- Can be parallelised and made more robust to local optima (particle swarm optimisation).
- In the context of probabilities basically just trying to integrate over parameter sets to find max-likelihood.

Intuitions

- Overall: Bayesian probabilities provide a framework for working with uncertainty.
- In particular, having uncertain hypotheses maps conceptually well with automated pattern recognition.
- Lots of sampling and search techniques are basically just trying to avoid doing integrals.

Exeunt

- Corpus is king.
- Uncertainty + priors implies parameterisation.
- Large amounts of ML is choosing good priors and heuristics for assumed latent distributions.
- Doesn't actually require a lot of math.
- It is lots of fun!
- Can we start being more data-directed at Xero? No more hard coded defaults or rules!

Thanks

Thanks!

- References:
    - Bishop — Pattern Recognition and Machine Learning
    - Norvig — Artificial Intelligence: A Modern Approach
    - Data sets: UCI Machine Learning Repository