COMP90051 Statistical Machine Learning
Semester 2, 2015

Workshop Week 2: Introduction; Bayesian Basics; Linear Models

July 30, 2015

This week we will mainly use R for workshop exercises, similarly for most workshops. However it is worth exploring in your own time

- R (possibly RStudio - a nice IDE for R), which we will use for some of our workshops, and which is popular in industry & research.
- Anaconda: A Python distribution for large-scale data processing, predictive analytics, and scientific computing. It includes numpy, scipy, matplotlib, sklearn and etc.
- Matlab with its many ML-related toolboxes

We won't dedicate time to teach these tools; the best way to learn is simply to use them. After perhaps starting with an online tutorial. Your tutors would be happy to point you to resources, if you can't find them.

1 Bayesian Posterior Updates

For this first hands-on workshop, use R, Matlab or Python (with Numpy/Scipy) (or a combination!). See how many of the following you can get through. Any you don't get to in class, try them later - feel free to ask follow-up questions on LMS.

1. Refer to the handout from lecture 2, the Bayesian update example. There we had r.v.'s X, μ where $P(X|\mu) \sim Normal(\mu, 1)$ and the prior was $P(\mu) \sim Normal(0, 1)$. We showed that after observing one $X = x$ the posterior becomes $P(\mu|X = x) \sim Normal(0.5x, 0.5)$. The more general update rule for $P(\mu) \sim Normal(a, b)$ is that $P(\mu|X = x) \sim Normal\left(\frac{b^2x+a}{b^2+1}, \frac{b^2}{b^2+1}\right)$.
2. Simulate training this model on streaming data coming from a process that is likely under the prior such as i.i.d $X_1, \ldots, X_n \sim Normal(0, 1)$. Sample the training data. Then starting with the prior, and going through each posterior, plot the successive distributions of μ. What's going on here? [Tip: you might overlay the densities with a spectrum of colours going from red to blue for example]
3. Repeat this experiment for a generating process that matches our main modeling assumptions, data that comes from a normal, but choose a mean that is unlikely under the prior, such as $X_1, \ldots, X_n \sim Normal(10, 1)$. What is similar/different in this scenario?

2 Linear Approaches

1. Take a look at the Orange tree growth data [1], which is automatically available as the Orange data.frame in R (no loading packages necessary). The data contains multiple measurements of trees (the kind made of wood; in rows), with the tree identifier (first column), tree age (days; second column), and tree circumferences (mm; third column).

2. Fit a linear regression predicting circumference based on age. Plot the data, and overlay the model. Calculate the fit's mean-squared error.

3. Load the 2D synthetic classification data [2], available as synth.tr when the MASS package is loaded via library(MASS). Instances are points in the 2D plane (first two columns "xs" and "ys") and the labels are classes ("0" or "1"; column "yc").

4. Fit a logistic regression. Plot the data, perhaps with points of each class with different marks (eg dots and crosses). Colour the points with their predicted class. Or if you prefer, plot the decision boundary. What is the classifier's accuracy?

5. Given time (or to play around with later) you might also try to

   (a) Derive a confusion matrix for the logistic regression using
   glm.pred=ifelse(predict(glm.fit,type="response")>0.5,"a","b")
   table(synth.tr$yc,glm.pred)

   (b) Plot an ROC curve using the ROCR package

   (c) Use the cv.glm function on synth.tr to derive 10-fold cross validation results

References

[1] N. R. Draper and H. Smith. *Applied Regression Analysis*. Wiley, 3rd edition, 1998.

[2] B. D. Ripley. Neural networks and related methods for classification (with discussion). *Journal of the Royal Statistical Society series B*, 56:409–456, 1994.