MLPR Tutorial[1] Sheet 3

Reminders: Attempt the tutorial questions, and ideally discuss them, before your tutorial. You can seek clarifications and hints on the class forum. Full answers will be released.

This week has less linear algebra! Try to spend some time preparing clear explanations.

---

1. **A Gaussian classifier:**

   A training set consists of one-dimensional examples from two classes. The training examples from class 1 are $\{0.5, 0.1, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.35, 0.25\}$ and the examples from class 2 are $\{0.9, 0.8, 0.75, 1.0\}$.

   a) Fit a one-dimensional Gaussian to each class by matching the mean and variance. Also estimate the class probabilities π_1 and π_2 by matching the observed class fractions. (This procedure fits the model with maximum likelihood: it selects the parameters that give the training data the highest probability.) Sketch the scores $p(x, y) = P(y)\, p(x \mid y)$ for each class y, as functions of input location x.

   b) What is the probability that the test point $x = 0.6$ belongs to class 1? Mark the decision boundary/ies on your sketch, the location(s) where $P(\text{class } 1 \mid x) = P(\text{class } 2 \mid x) = 0.5$. You are not required to calculate the location(s) exactly.

   c) Are the decisions that the model makes reasonable for very negative x and very positive x? Are there any changes we could consider making to the model if we wanted to change the model's asymptotic behaviour?

2. **Gradient descent:**

   Let $E(\mathbf{w})$ be a differentiable function. Consider the gradient descent procedure

   $$\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} - \eta \nabla_{\mathbf{w}} E.$$

   a) Are the following true or false? Prepare a clear explanation, stating any necessary assumptions:

      i) Let $\mathbf{w}^{(1)}$ be the result of taking one gradient step. Then the error always improves, i.e., $E(\mathbf{w}^{(1)}) \leq E(\mathbf{w}^{(0)})$.

      ii) There exists some choice of the step size η such that $E(\mathbf{w}^{(1)}) < E(\mathbf{w}^{(0)})$.

   b) A common programming mistake is to forget the minus sign in either the descent procedure or in the gradient evaluation. As a result one unintentionally writes a procedure that does $\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} + \eta \nabla_{\mathbf{w}} E$. What happens?

3. **Maximum likelihood and logistic regression:**

   Maximum likelihood logistic regression maximizes the log probability of the labels,

   $$\sum_n \log P(y^{(n)} \mid \mathbf{x}^{(n)}, \mathbf{w}),$$

   with respect to the weights $\mathbf{w}$. As usual, $y^{(n)}$ is a binary label at input location $\mathbf{x}^{(n)}$.

   The training data is said to be *linearly separable* if the two classes can be completely

1. Parts of this tutorial sheet are based on previous versions by Amos Storkey, Charles Sutton, and Chris Williams

separated by a hyperplane. That means we can find a decision boundary

$$P(y^{(n)}=1 \mid \mathbf{x}^{(n)}, \mathbf{w}, b) = \sigma(\mathbf{w}^\top \mathbf{x}^{(n)} + b) = 0.5, \qquad \text{where } \sigma(a) = \frac{1}{1+e^{-a}},$$

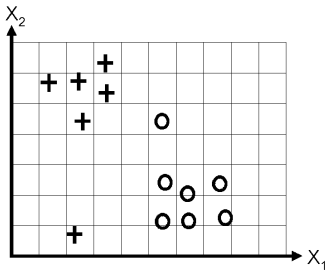
such that all the $y=1$ labels are on one side (with probability greater than 0.5), and all of the $y \neq 1$ labels are on the other side.

a) Show that if the training data is linearly separable with a decision hyperplane specified by $\mathbf{w}$ and b, the data is also separable with the boundary given by $\tilde{\mathbf{w}}$ and $\tilde{b}$, where $\tilde{\mathbf{w}} = c\mathbf{w}$ and $\tilde{b} = cb$ for any scalar $c > 0$.

b) What consequence does the above result have for maximum likelihood training of logistic regression for linearly separable data?

4. **Logistic regression and maximum likelihood:** (Murphy, Exercise 8.7, by Jaaakkola.)

Consider the following data set:



a) Suppose that we fit a logistic regression model with a bias weight w_0, that is $p(y=1 \mid \mathbf{x}, \mathbf{w}) = \sigma(w_0 + w_1 x_1 + w_2 x_2)$, by maximum likelihood, obtaining parameters $\hat{\mathbf{w}}$. Sketch a possible decision boundary corresponding to $\hat{\mathbf{w}}$. Is your answer unique? How many classification errors does your method make on the training set?

b) Now suppose that we regularize only the w_0 parameter, that is, we minimize

$$J_0(\mathbf{w}) = -\ell(\mathbf{w}) + \lambda w_0^2,$$

where ℓ is the log-likelihood of $\mathbf{w}$ (the log-probability of the labels given those parameters).

Suppose λ is a very large number, so we regularize w_0 all the way to 0, but all other parameters are unregularized. Sketch a possible decision boundary. How many classification errors does your method make on the training set? Hint: consider the behaviour of simple linear regression, $w_0 + w_1 x_1 + w_2 x_2$ when $x_1 = x_2 = 0$.

c) Now suppose that we regularize only the w_1 parameter, i.e., we minimize

$$J_1(\mathbf{w}) = -\ell(\mathbf{w}) + \lambda w_1^2,$$

Again suppose λ is a very large number. Sketch a possible decision boundary. How many classification errors does your method make on the training set?

d) Now suppose that we regularize only the w2 parameter, i.e., we minimize

$$J_2(\mathbf{w}) = -\ell(\mathbf{w}) + \lambda w_2^2,$$

Again suppose λ is a very large number. Sketch a possible decision boundary. How many classification errors does your method make on the training set?