CS540 Spring 2010: homework 4

1 Logistic regression vs LDA/QDA

(Source: Jaakkola)
Suppose we train the following binary classifiers via maximum likelihood.

1. GaussI: A generative classifier, where the class conditional densities are Gaussian, with both covariance matrices set to $\mathbf{I}$ (identity matrix), i.e., $p(\mathbf{x}|y=c) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_c, \mathbf{I})$. We assume $p(y)$ is uniform.
2. GaussX: as for GaussI, but the covariance matrices are unconstrained, i.e., $p(\mathbf{x}|y=c) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$.
3. LinLog: A logistic regression model with linear features.
4. QuadLog: A logistic regression model, using linear and quadratic features (i.e., polynomial basis function expansion of degree 2).

After training we compute the performance of each model M on the training set as follows:

$$L(M) = \frac{1}{n}\sum_{i=1}^{n} \log p(y_i|\mathbf{x}_i, \hat{\boldsymbol{\theta}}, M) \tag{1}$$

(Note that this is the *conditional* log-likelihood $p(y|\mathbf{x}, \hat{\boldsymbol{\theta}})$ and not the joint log-likelihood $p(y, \mathbf{x}|\hat{\boldsymbol{\theta}})$.) We now want to compare the performance of each model. We will write $L(M) \leq L(M')$ if model M *must* have lower (or equal) log likelihood (on the training set) than M', for any training set (in other words, M is worse than M', at least as far as training set logprob is concerned). For each of the following model pairs, state whether $L(M) \leq L(M')$, $L(M) \geq L(M')$, or whether no such statement can be made (i.e., M might sometimes be better than M' and sometimes worse); also, for each question, briefly (1-2 sentences) explain why.

1. GaussI, LinLog.
2. GaussX, QuadLog.
3. LinLog, QuadLog.
4. GaussI, QuadLog.
5. Now suppose we measure performance in terms of the average misclassification rate on the training set:

   $$R(M) = \frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}(\mathbf{x}_i)) \tag{2}$$

   Is it true in general that $L(M) > L(M')$ implies that $R(M) < R(M')$? Explain why or why not.

2 Decision boundary for LDA with semi tied covariances

Consider a generative classifier with class conditional densities of the form $\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$. In LDA, we assume $\boldsymbol{\Sigma}_c = \boldsymbol{\Sigma}$, and in QDA, each $\boldsymbol{\Sigma}_c$ is arbitrary. Here we consider the 2 class case in which $\boldsymbol{\Sigma}_1 = k\boldsymbol{\Sigma}_0$, for $k > 1$. That is, the Gaussian ellipsoids have the same "shape", but the one for class 1 is "wider". Derive an expression for $p(y=1|\mathbf{x}, \boldsymbol{\theta})$, simplifying as much as possible. Give a geometric interpretation of your result, if possible.

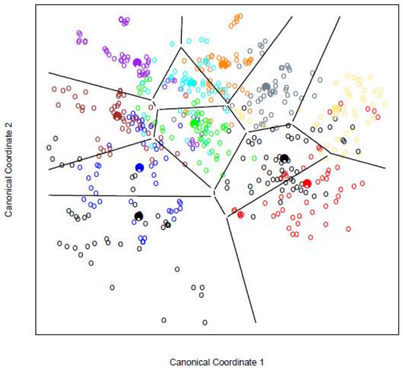

Figure 1: Projection of 11-class 10-dimensional vowel data to 2d using Fisher's LDA. The dark black circles are the class means. Source: Figure 4.11 of [HTF09].

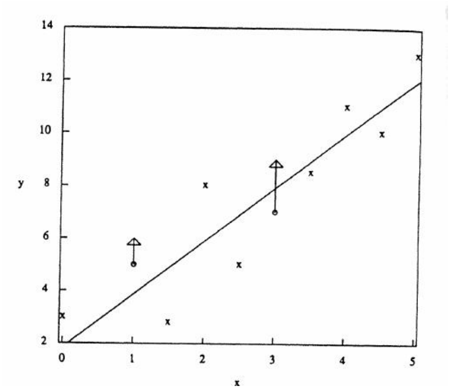

Figure 2: Example of right-censored data. The circles are the censored points, the arrow heads are the imputed values, and the line is the line fit to the imputed and uncensored data. Source: Figure 5.6 of [Tan96]

3 Fisher's LDA applied to vowel data

1. Consider the vowel data from [HTF09]. The training data is available (as a text file) from `http://www-stat.stanford.edu/~tibs/ElemStatLearn/datasets/vowel.train`. You can use the `dlmread` function to parse this file. Use the following code fragment to compute a 2d dimensional projection of the data: `[B, Z] = pcaPmtk(Xtrain, 2).` Plot the projected data `Z`, color coded by class. Compute the class conditional means $\boldsymbol{\mu}_c \in \mathbb{R}^{10}$ and plot their 2d projection too. Turn in your code and plot.

2. Implement the multi-class version of Fisher's LDA. You may assume $\mathbf{S}_W$ is invertible, and hence you just need to find the first K eigenvectors of $\mathbf{S}_W^{-1}\mathbf{S}_B$. Project the data and the means projected onto this 2d subspace. The result should look similar to Figure 1. (At least I think it should: the details on how this figure was created are not given in on [HTF09, p118]. Note that this book is available online for free from `http://www-stat.stanford.edu/~tibs/ElemStatLearn/download.html` if you want to read more about what they did.) Turn in your code and plot. (You can use `plotDecisionBoundary` to get the boundaries.)

4 EM for censored linear regression

This exercise is about the EM algorithm for censored linear regression discussed in section 2.1 of the EM handout `https://people.cs.ubc.ca/~murphyk/passwordProtected/handoutEmRegClassif.pdf`.

Note that there is an error in equation 10 of that handout. The correct result is

$$\mathbb{E}\left[z_i|\boldsymbol{\theta}, z_i \geq c_i\right] \quad = \quad \mu_i + \sigma H(\frac{c_i - \mu_i}{\sigma}) \tag{3}$$

where we have defined

$$H(u) \overset{\text{def}}{=} \frac{\phi(u)}{1 - \Phi(u)} \tag{4}$$

1. Show that

$$\mathbb{E}\left[z_i^2|\boldsymbol{\theta}, z_i \geq c_i\right] \quad = \quad \mu_i + \sigma^2 + \sigma(c_i + \mu_i) H\left(\frac{c_i - \mu_i}{\sigma}\right) \tag{5}$$

2. Derive the M step for σ^2.

3. Implement the algorithm and apply it to the Schmee and Hahn data discussed in section 2.2 of th EM handout. That is, use a linear regression model to predict $y_i = \log_{10}(\text{i'th failure time})$, using covariate $x_i = 1000/(\text{temperature at 273.2})$. What are the estimated parameters? As a sanity check, [Tan96, p69] got the following results, after 16 iterations of EM:

$$\hat{w}_0 = -6.019, \hat{w}_1 = 4.311, \hat{\sigma} = 0.2592 \tag{6}$$

4. Plot the data and the censored and predicted values (on the same axes), giving a result similar to Figure 2. Plot the line fit by EM, as well as the line fit to the raw data (ignoring the fact that some values were censored).

Turn in your code, numbers and plot.

5 EM for robust regression with a Student T noise model

This exercise is about the EM algorithm for robust linear regression discussed in section 3.1.1 of the EM handout `https://people.cs.ubc.ca/~murphyk/passwordProtected/handoutEmRegClassif.pdf`. (More details can be found in [LLT89].)

1. Implement the EM algorithm assuming the dof ν is fixed. (Weighted least squares is discussed on pdf p314 of my book.)

2. Apply it to `linregRobustDemo`. How do your results compare to using gradient descent?

3. Apply your method to the famous stack loss data shown in Figure 3(a). That is, fit a model of the form

$$y_i = \beta_0 + \sum_j \beta_j x_{ij} + \epsilon_i \tag{7}$$

Try the following values for the dof: $\nu \in \{\infty, 8, 4, 3, 2, 1.1, 1, 0.5\}$. For each such value, compute the log-likelihood on the training set, and the print the MLE of the parameters. Your results should be similar to Figure 3(b).

References

[HTF09] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, 2009. 2nd edition.

[LLT89] K. Lange, R. Little, and J. Taylor. Robust statistical modeling using the t disribution. *J. of the Am. Stat. Assoc.*, 84(408):881–896, 1989.

[MK97] G. J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley, 1997.

[Tan96] M. Tanner. *Tools for statistical inference*. Springer, 1996.

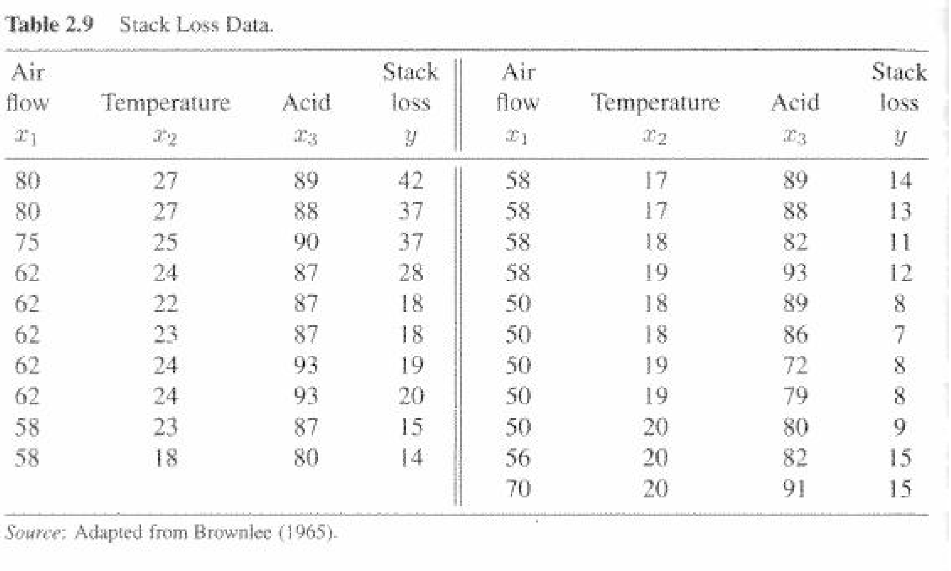

Table 2.9 Stack Loss Data.

Air flow x_1	Temperature x_2	Acid x_3	Stack loss y	Air flow x_1	Temperature x_2	Acid x_3	Stack loss y
80	27	89	42	58	17	89	14
80	27	88	37	58	17	88	13
75	25	90	37	58	18	82	11
62	24	87	28	58	19	93	12
62	22	87	18	50	18	89	8
62	23	87	18	50	18	86	7
62	24	93	19	50	19	72	8
62	24	93	20	50	19	79	8
58	23	87	15	50	20	80	9
58	18	80	14	56	20	82	15
				70	20	91	15

Source: Adapted from Brownlee (1965).

(a)

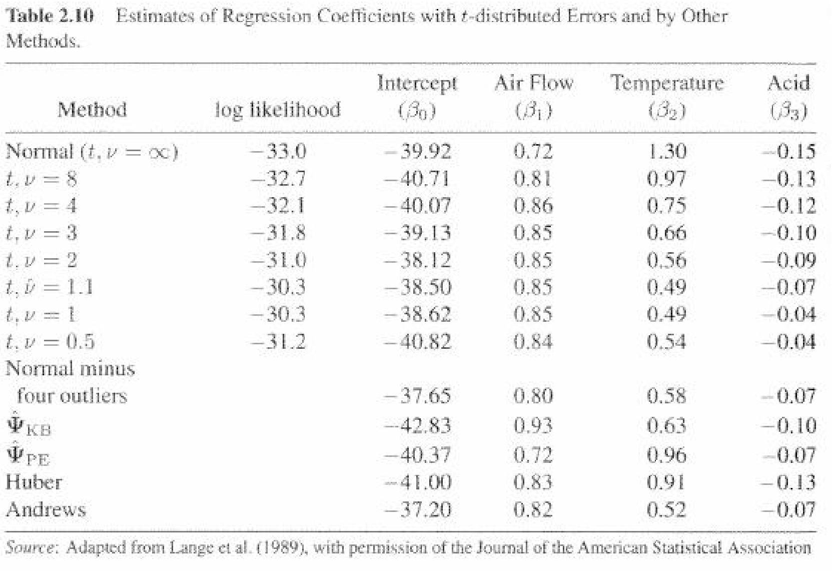

Table 2.10 Estimates of Regression Coefficients with t-distributed Errors and by Other Methods.

Method	log likelihood	Intercept (β_0)	Air Flow (β_1)	Temperature (β_2)	Acid (β_3)
Normal $(t, \nu = \infty)$	−33.0	−39.92	0.72	1.30	−0.15
$t, \nu = 8$	−32.7	−40.71	0.81	0.97	−0.13
$t, \nu = 4$	−32.1	−40.07	0.86	0.75	−0.12
$t, \nu = 3$	−31.8	−39.13	0.85	0.66	−0.10
$t, \nu = 2$	−31.0	−38.12	0.85	0.56	−0.09
$t, \hat{\nu} = 1.1$	−30.3	−38.50	0.85	0.49	−0.07
$t, \nu = 1$	−30.3	−38.62	0.85	0.49	−0.04
$t, \nu = 0.5$	−31.2	−40.82	0.84	0.54	−0.04
Normal minus four outliers		−37.65	0.80	0.58	−0.07
$\hat{\Psi}_{\mathrm{KB}}$		−42.83	0.93	0.63	−0.10
$\hat{\Psi}_{\mathrm{PE}}$		−40.37	0.72	0.96	−0.07
Huber		−41.00	0.83	0.91	−0.13
Andrews		−37.20	0.82	0.52	−0.07

Source: Adapted from Lange et al. (1989), with permission of the Journal of the American Statistical Association

(b)

Figure 3: (a) Stack loss data. Source: Table 2.9 of [MK97]. (b) Results on this data using various models. Source: Table 2.10 of [MK97].