Model Building and Goodness of Fit:
(Case Study: Logistic Regression)
Reminder: Logistic Regression

• Logistic Regression:
  • \[ \log \left( \frac{\pi_i}{1 - \pi_i} \right) = \beta_0 + \beta_1 x_1^{(i)} + \cdots + \beta_k x_k^{(i)} \]
  • \(\pi_i\): the likelihood that \(y^{(i)} = 1\)

• (Log-)Likelihood:
  • Compute \(\pi_i\) for every datapoint for which \(y^{(i)} = 1\), and \((1 - \pi_i)\) for every datapoint for which \(y^{(i)} = 0\).
  • If the fit is very good, the product
    \[ P(y|\beta) = \prod_{i=1}^{n} \pi_i^{y_i} (1 - \pi_i)^{1-y_i} \]
    is close to 1
  • \[ \log P(y|\beta) = \sum_{i=1}^{n} (y_i \log \pi_i + (1 - y_i) \log(1 - \pi_i)) \]
    is close to 0
Deviance

\[ deviance = const - 2 \log P(y|\beta) \]

• (where \( \beta \) is the fitted parameter – the one that maximizes \( \log P(y|\beta) \). In other words, \( \log P(y|\beta) = LMAX \))

• Smaller deviance => better fit
  • “Better fit” means \( \pi_i \) is close to 1 if \( y_i \) is close to 1, and \( \pi_i \) is close to 0 if \( y_i \) is close to 0
Last Time

• Null Hypothesis: the extra coefficients in the full model are 0

• Test Statistic:
  • \( LRT = 2 \log(LMAX_{full}) - 2 \log(LMAX_{reduced}) \)
    • Has a \( \chi^2 \) distribution with df= (#of extra parameters in the full model)

• Test?
Last Time

• Null Hypothesis: the extra coefficients in the full model are 0

• Test Statistic:
  • \( LRT = 2 \log(LMAX_{full}) - 2 \log(LMAX_{reduced}) \)
  • Has a \( \chi^2 \) distribution with \( df=(\#\text{of extra parameters in the full model}) \)

• Test?
  • \((1-pchisq(L, \text{df}=df))\)
Which Covariates to Include?

• \( AIC = \text{Deviance} + 2 \times p \)
  • \( p \) – number of parameters
  • Popularly called “Akaike Information Criterion”
  • Hirutogu Akaike calls it “An Information Criterion”

• Smaller AIC => Better Model
  • Note: larger model always implies smaller deviance
    • Problem: Why?
  • AIC compensates for that
    • The full model has to be reduce the deviance by enough to be considered better than the reduced model
Classification (Iris Example)
Visualization

• (in R)
Classification

• Classification:
  • Given data, we want to predict the class of the datapoint

• Fit a logistic model and pick a cut point
  • Default: 0.5

• If $\hat{\pi}^* > 0.5$, predict $y^* = 1$
• If $\hat{\pi}^* < 0.5$, predict $y^* = 0$
# Confusion Matrix

<table>
<thead>
<tr>
<th></th>
<th>+</th>
<th>-</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>TP</td>
<td>FN</td>
<td>Sensitivity = TP / (TP + FN)</td>
</tr>
<tr>
<td>Dc</td>
<td>FP</td>
<td>TN</td>
<td>Specificity = TN / (TN + FP)</td>
</tr>
</tbody>
</table>

- TP: True Positive
- FN: False Negative
- FP: False Positive
- TN: True Negative

- PPV = TP / (TP + FP)
- NPV = TN / (TN + FN)
ROC Curves

• Receiver Operating Characteristic
• A plot of sensitivity vs. specificity (complement)
• Originally designed to grade radar detection methods for German planes
• Decades later, their usefulness in classification problems was realized
  • But the name stuck
**ROC Curve**

```r
require(pROC)

> titan.roc <- with(titanR, roc(Isurvived, p, percent=T, auc=T, plot=T, auc.polygon=T, max.auc.polygon=T, print.auc=T, main= "ROC curve"))
```

![ROC curve](image)

AUC: 77.3%
AUC

- Area Under the Curve
  - Larger is better
  - Ideally: 100% sensitivity for any specificity