Lecture Nine – Confusion Matrix
Checking the accuracy of the Logistic Regression Model
Professor Esfandiari

A confusion matrix is a table that is often used to **describe the performance of a classification model** (or "classifier") on a set of test data for which the true values are known. The confusion matrix itself is relatively simple to understand, but the related terminology can be confusing.

We will start with an **example confusion matrix for a binary classifier**. In the following you are given the data for 200 patients that have diabetic. Based on a logistic regression model, we have made predictions. The following table presents the actual values and the predicted values.

Actual	Predicted		Colum totals
	No	Yes	
No	60 TRUE NEGATIVE	12 FALSE POSITIVE TYPE I ERROR	72
Yes	8 FALSE NEGATIVE TYPE II ERROR	120 TRUE POSITIVE	128
Row totals	68	132	200

Based on the prediction model,
Of the two hundred patients, 132 were classified as having the disease and 68 were classified as not having the disease.

Based on the actual results,
Of the two hundred patients, 128 actually had diabetic II and 72 did not.

We will now define, the different cells of the confusion matrix
TRUE POSITIVE
True positive are the cases in which the patient actually has the disease and the model also predicts that the patient has the disease.

FALSE POSITIVE
False positive are the cases in which the patient actually has the disease, but, the model predicts that the patient does has the disease. **This is known as TYPE I ERROR.**

FALSE NEGATIVE
False negative are the cases in which the patient actually does not have the disease, but, the model predicts that the patient has the disease. **THIS IS KNOWN AS TYPE II ERROR.**

TRUE POSITIVE
False positive are the cases in which the patient actually has the disease and the model predicts that the patient has the disease.

Other Concepts Underlying the Model

ACCURACY

Overall, how well did the prediction based on the logistic model work?

Accuracy = $\frac{True\ positive+Ture\ negative}{Total\ sample}$ **=** $\frac{120+60}{200}$ **= 180/200 = 0.90**

This means that in 90% of the cases the prediction based on the logistic model makes the correct classification.

MISCLASSIFICATION RATE – ERROR RATE
Overall, how often is the classification done by the prediction model wrong?

Misclassification Rate = $\frac{false\ positive+false\ negative}{Total\ sample}$ **=** $\frac{12+8}{200}$ **= 20/200 = 0.10**

Misclassification Rate = 1- accuracy rate = 1 – 0.90 = 0.10

This means that in 10% of the cases the prediction based on the logistic model makes the incorrect classification.

TRUE POSITIVE RATE – SENSITIVITY RATE
True positive rate: When the patient "**actually has diabetic II**", and the **model also predicts "yes".**

=TP/actual yes
= 120/128 = 0.9375

FALSE POSITIVE RATE – SENSITIVITY RATE
False positive rate: When the patient "**does not have diabetic II**", and **model predicts** "**yes**".

=FP/Actual No
= 12/72 = 0.167

- **Specificity:** When it's actually no, how often does it predict no?

  Specificity = $\frac{True\ No}{Actual\ No}$ **= 60/72 = 0.833**

  equivalent to 1 minus False Positive Rate = 1- 0.167 = 0.833

- **Precision:** When it predicts yes, how often is it correct?

  Precision = $\frac{True\ positive}{Predicted\ yes}$ **= 120/132 = 0.9090909**

- **Prevalence:** How often does the yes condition actually occur in our sample?

  actual yes/total = 128/200 = 0.64

Calculation of the Confusion Matrix using R

Using the diabetic data, we will create a logistic regression model for the prediction of having diabetic type II from age, smoking, and hypertension.

```
> m1<-glm(Diabetes.new~Age+SmokingStatus_NISTCode+HypertensionDX,
family="binomial")
> summary(m1)


Coefficients:
                                  Estimate   Std. Error z value  Pr(>|z|)
(Intercept)                       -3.380674  0.174617 -19.361   < 2e-16 ***
Age                               0.025801   0.002736   9.430   < 2e-16 ***
SmokingStatus_NISTCodeFORMER -0.355067   0.085528  -4.151   3.3e-05 ***
SmokingStatus_NISTCodeTRUE        -0.334447   0.130085  -2.571    0.0101 *
HypertensionDXyes                 1.129735   0.090237  12.520   < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 4404.9  on 4402  degrees of freedom
Residual deviance: 3953.5  on 4398  degrees of freedom
  (5545 observations deleted due to missingness)
AIC: 3963.5

Number of Fisher Scoring iterations: 5
```

We will now use R to create the confusion matrix

R Codes

```
> library(nnet)
> m1<-
multinom(Diabetes.new~Age+HypertensionDX+SmokingStatus_NISTCode,diabetic)

# weights:  6 (5 variable)
initial  value 3051.927036
iter  10 value 1976.747412
final  value 1976.731777
converged
```

R codes for the creation of the confusion matrix continued

```
> p<-predict(m1,diabetic)
> tab<-table(p,diabetic$Diabetes.new)
> tab

  p     0     1
0    3517  876
1    6     4
```

Predicted values	Actual Value		Row totals
	Does not have diabetic	Has diabetic	
Does not have diabetic	3517 True negative	876 False Positive	4393
Has Diabetic	6 False Negative	4 True positive	10
Column Total	3523	880	4403

- **Accuracy:** Overall, how often is the classifier (predicted model) correct?
  (TP+TN)/total = (4+3517)/4403 = **0.799682**
- **Misclassification Rate or error rate:** Overall, how often is the classifier wrong?
  (FP+FN)/total = (876+6)/4403 = **0.200318**
  equivalent to 1 minus Accuracy
- **True Positive Rate:** When it's actually yes, how often does it predict yes?
  TP/actual yes = 4/880 = **0.004545455**
  also known as "Sensitivity"
- **False Positive Rate:** When it's actually no, how often does it predict yes?
  FP/actual no = 876/3523 = **0.2486517**
- **Specificity:** When it's actually no, how often does it predict no?
  TN/actual no = 3517/3523 **= 0.9982969**
  equivalent to 1 minus False Positive Rate
- **Precision:** When it predicts yes, how often is it correct?
  TP/predicted yes = 4/10 = **0.4**
- **Prevalence:** How often does the yes condition actually occur in our sample?
  actual yes/total = 880/4403 = **0.1998637**

CONCLUSION: **Based on the confusion matrix given above, the accuracy rate is 80%. So, the predictive power of the model is good.** You want most of the points to be on the diagonal; that is true positive and true negative. The model seems to be doing well with respect to true negative but not so well with respect to true positive.