1 Diagnostics of Classifiers

1.1 Confusion Matrix

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

1.2 Accuracy

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

1.3 True Positive Rate

$$\text{TPR} = \frac{TP}{TP + FN}$$

1.4 False Positive Rate / Type I Error Rate

$$\text{FPR} = \frac{FP}{FP + TN}$$

1.5 False Negative Rate / Type II Error Rate

$$\text{FNR} = \frac{FN}{TP + FN}$$

1.6 True Negative Rate

$$\text{TNR} = \frac{TN}{TN + FP}$$

1.7 Precision

$$\text{Precision} = \frac{TP}{TP + FP}$$

1.7.1 Remarks

1. Precision is useful when costly actions will be followed up on the data predicted to be positive,

2. because precision gives the proportion of actual positives among those predicted to be positive

3. For example, if an insurance company wants to predict potential customers interested in purchasing insurance, and the cost to try to sell an insurance to a potential customer is non-trivial (e.g. insurance agent has to house visit the customer).

1.8 N-Fold Cross Validation

1.8.1 Algorithm

1. The entire dataset is randomly split into N datasets of approximately equal size.
2. $N-1$ of these datasets are treated as the training dataset, while the remaining one is the test dataset. A measure of the model error is obtained.
3. This process is repeated across the various combinations of N datasets taken $N-1$ at a time.
4. The observed N models errors are averaged across the N folds

1.9 ROC Curve (TPR vs FPR Trade-off)

1. Graph of True Positive Rate (TPR) against False Positive Rate (FPR)
2. As TPR increases, FPR tend to increase as well
   (a) Increasing TPR may be a double-edged sword
3. TPR = FPR = 0 means binary classifier classifies everything as negative
4. TPR = FPR = 1 means binary classifier classifies everything as positive

1.10 Bias-Variance tradeoff

1. error = bias^2 + variance + irreducible error
2. As variance increases, bias decreases, and vice versa

1.11 Calculation Intensive Exam Question & Solution

Midterm Q6. Consider the following confusion matrix for a classifier

		Predicted Class	
		Positive	Negative
Actual Class	Positive	20	75
	Negative	140	55

The false negative rate (FNR) of the classifier is ____ (round to 3 decimal places).

Solution

1. Copy paste the following code:

```
gcmfv <- function(tp, fn, fp, tn) {
  # generates confusion matrix from values
  return (matrix(c(tp, fn, fp, tn), nrow = 2, ncol = 2, byrow = TRUE))
}

tp <- function(m) {
  # true positive from confusion matrix m
  return (m[1, 1])
}

fn <- function(m) {
  # false negative from confusion matrix m
  return (m[1, 2])
}

fp <- function(m) {
  # false positive from confusion matrix m
  return (m[2, 1])
}

tn <- function(m) {
  # true negative from confusion matrix m
  return (m[2, 2])
}

accuracy <- function(m) {
  return ((tp(m)+tn(m))/(tp(m)+tn(m)+fp(m)+fn(m)))
}

tpr <- function(m) {
  return (tp(m)/(tp(m)+fn(m)))
}

fpr <- function(m) {
  return (fp(m)/(fp(m)+tn(m)))
}

fnr <- function(m) {
```

```r
  return (fn(m)/(fn(m)+tp(m)))
}

tnr <- function(m) {
  return (tn(m)/(tn(m)+fp(m)))
}

precision <- function(m) {
  return (tp(m)/(tp(m)+fp(m)))
}
```

2. Create Confusion Matrix

```r
confusion.matrix <- gcmfv(tp=20, fn=75, fp=140, tn=55)
```

3. Get the metric you need

```r
fnr(confusion.matrix)


## [1] 0.7894737
```