Machine Learning

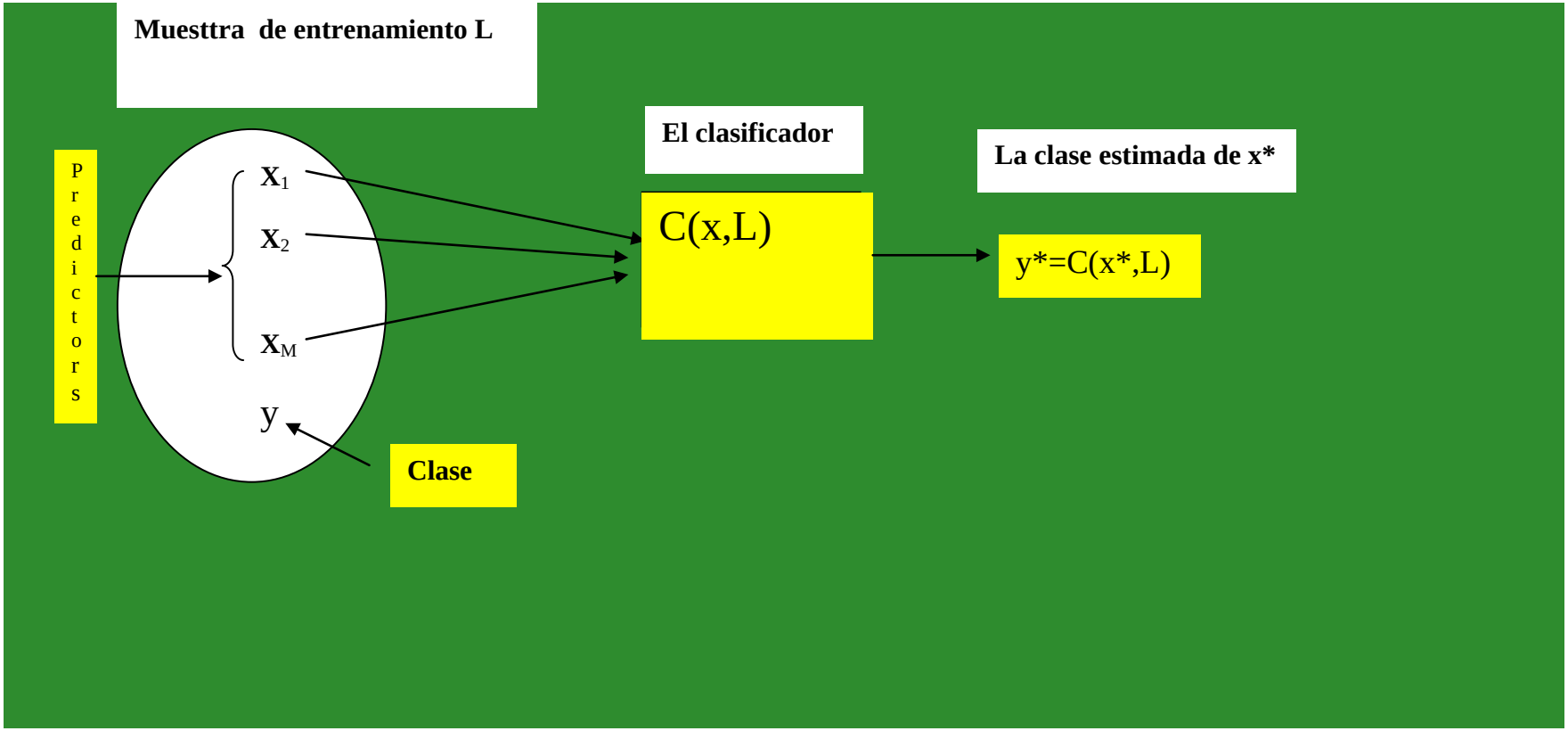
Clasificación Supervisada

Dr. Edgar Acuña
Departamento de Ciencias Matemáticas
Universidad de Puerto Rico-Mayaguez

E-mail: edgar.acuna@upr.edu, eacunaf@gmail.com

Website:academic.uprm.edu/eacuna

El problema de clasificacion supervisada



Clasificacion Supervisada—un proceso de dos etapas.

- Construccion del modelo:
  - Se asume que cada objeto pertenece a una clase predefinida, segun es definido en la columna de clases.
  - El modelo es representado por reglas de clasificacion, arboles de decision, o formulas matematicas.
- Uso del modelo, para clasificar futuros objetos.
  - Estimar la precision del modelo
    - La columna de clases de los objetos de la muestra de prueba es comparado con las clases predichas por el modelo. La muestra de prueba debe ser independiente de la muestra de entrenamiento de lo contrario puede ocurrir “sobreajuste”.
    - La tasa de precision es el porcentaje de objetos de la muestra de prueba que son correctamente clasificadas por el modelo.

Clasificacion Supervisada vs. Regresion

- Clasificacion Supervisada:
  - Predice valores categoricos que representan clases.
  - Construye un modelo basado en la muestra de entrenamiento y lo usa para clasificar nuevos datos.
- Regresion:
  - Modela funciones de valores continuous. Predice valores desconocidos o valores faltantes.

Metodos de clasificacion supervisada

1. Analisis Discriminante Lineal.
2. Metodos No Lineales: Discriminacion Cuadratica, Regresion Logistica, Projection Pursuit.
3. Naive Bayes y Redes Bayesianas
4. K vecinos mas cercanos.
5. Clasificadores basados en estimacion de densidad por kernel. Clasificadores que usan mezclas Gausianas.
6. Clasificadores basados en reglas
7. Arboles de Decision.
8. Random Forest
9. Redes Neurales: El perceptron de multicapas, Funciones bases radiales, mapas auto-organizantes de Kohonen.
10. Maquinas de soporte vectorial.

Clasificacion supervisada desde un punto de vista Bayesiano

Suponga que conocemos de antemano la probabilidad apriori π_i (i=1, 2,…G) de que un objeto pertenezca a la clase C_i. Si no se conoce informacion adicional entonces un objeto cualquiera sera clasificado como perteneciente a la clase C_i si

$$\pi_i > \pi_j \text{ para } i=1,2,.....G,\ i \neq j \qquad (3.1)$$

Un forma facil de estimar la probabilidad a priori es tomarla como la frecuencia relativa de la clase. Sin embargo, usualmente alguna informacion adicional es conocida acerca del objeto, tal como un vector x de mediciones hechas en el objeto a ser clasificado. En este caso, se compara la probabilidad de pertenecer a cada clase para un objeto con vector de mediciones x, y el objeto es clasificado como de clase Ci si

$$P(C_i/\mathbf{x}) > P(C_j/\mathbf{x}) \text{ para todo } i \neq j \qquad (3.2)$$

Esta regla de decision es llamada la Regla de **error minimo**.

Notar que $i=\text{argmax}_k\ P(C_k/\mathbf{x})$ para todo k en 1,2….G.

Clasificacion supervisada desde un punto de vista Bayesiano

Probabilidad Condicional: Si A y B son dos elementos del mismo espacio muestral

$$P(A/B)=P(AyB)/P(B) \quad \text{si } P(B)>0$$

Regla del producto para probabilidades:

$$P(AyB)=P(A)P(B/A) \text{ o } P(B)P(A/B)$$

Regla de la Cadena para Probabilidades:

$$P(AyByC)=P(A)P(B/A)P(C/AyB)$$

Regla de Bayes:

$$P(A/B)=[P(A)P(B/A)]/P(B)$$

P(A) es llamada la probabilidad a priori

P(A/B) es llamada la probabilidad a posteriori

Clasificacion supervisada desde un punto de vista Bayesiano

En clasificación:

A: es el evento de que un objeto caiga es una clase dada.

P(A) usualmente es estimada por la frecuencia relativa de la clase

P(A/B) es la probabilidad de que un objeto caiga en una clase dada si tiene un vector de mediciones x

P(B/A) es la distribucion de probabilidades del vector aleatorio x en una clase dada

P(B) es la distribucion de probabilidades del vector aleatorio x sin importar las clases. NO interesa para los calculos

Clasificacion Bayesiana

Las probabilidades $P(C_i/x)$ son llamadas probabilidades posteriores. Desafortunadamente, estas probabilidades raras veces son conocidas y deben ser estimadas. Esto ocurre en regresion logistica, clasificadores por vecinos mas cercanos, y redes neurales.

Una formulacion mas conveniente de la regla anterior puede ser obtenida aplicando el teorema de Bayes, que afirma que

$$P(C_i / \mathbf{x}) = \frac{f(\mathbf{x}/C_i)\pi_i}{f(\mathbf{x})} \qquad (3.3)$$

donde, $f(\mathbf{x}/C_i)$, es la densidad condicional de la clase C_i. Por lo tanto un objeto sera clasificado en la clase C_i si

$$f(\mathbf{x}/C_i)\pi_i > f(\mathbf{x}/C_j)\pi_j \qquad (3.4)$$

para todo $i \neq j$. Esto es, $i=\text{argmax}_k f(x/C_k)\pi_k$

Si las densidades condicionales de clase, *f(**x**/ C_i) son conocidas entonces el problema de clasificacion esta resuelto explicitamente, como ocurre en el analisis discriminante lineal y cuadratico.*

Pero frecuentemente las *f(**x**/ C_i) no son conocidas y ellas deben ser estimadas usando la muestra de entrenamiento. Este es el caso de los clasificadores naive Bayes, k-nn, los clasificadores basados en estimacion de densidad por kernel y aquellos basados en mezclas Gaussianas.*

Analisis Discriminante Lineal

Considerar la siguiente muestra de entrenamiento con p atributos y dos clases

Y	X_1	X_2	…	X_p
1	X_{11}	X_{21}	….	X_{p1}
1	X_{12}	X_{22}	…	X_{p2}
..	..	..	..	..
1	X_{1n1}	X_{2n1}	…	X_{pn1}
2	$X_{1,n1+1}$	$X_{2,n1+1}$	…	$X_{p,n1+1}$
2	$X_{1,n1+2}$	$X_{2,n1+2}$	…	$X_{p,n1+2}$
..	..	..	…	..
2	$X_{1,n1+n2}$	$X_{2,n1+n2}$	…	$X_{p,n1+n2}$

Analisis discriminante lineal

Sea $\overline{x}_1$ el vector de medias de los p atributos en clase 1, y sea l $\overline{x}_2$ el vector de medias correspondente para la clase 2.

Considerar que μ_1 y μ_2 son los vectores de media de las respectivas clases poblacionales. Asumir que ambas poblaciones tienen la misma matriz de covarianza, i.e. $\Sigma_1=\Sigma_2=\Sigma$. Esta es conocida como la propiedad de homocedasticidad.

Por ahora, no necesitamos asumir que el vector aleatorio de predictoras **x**=(x1,…..xp) esta normalmente distribuido.

Discriminacion lineal esta basado en el siguiente hecho: una instancia (objeto) x es asignado a la clase C1 si

$$D(\mathbf{x}, C1)<D(\mathbf{x},C2) \quad (2.1)$$

donde $D(\mathbf{x},Ci)=(\mathbf{x}-\mu_i)'\Sigma^{-1}(\mathbf{x}-\mu_i)$, for i=1,2, representa el cuadrado de de la distancia ***squared Mahalanobis de x*** *al centro de la clase* C_i.

Analisis discriminante lineal (cont)

La expresion (2.1) puede ser escrita como

$$(\mu_1 - \mu_2)'\Sigma^{-1}[x - (1/2)(\mu_1 + \mu_2)] > 0 \qquad (2.2)$$

Usando la muestra de entrenamiento, μ_i puede ser estimada por y $\bar{x}_i$ y $\mathbf{\Sigma}$ es estimada por S, la matriz de covarianza combinada, la cual es calculada por

$$S = \frac{(n_1 - 1)S_1 + (n_2 - 1)S_2}{n_1 + n_2 - 2}$$

donde, S1 y S2 representan las matrices de covarianza muestrales del vector de predictoras en cada clase. Luego, la version muestral de (2.2) esta dada por

$$(\bar{X}_1 - \bar{X}_2)'S^{-1}[x - (1/2)(\bar{X}_1 + \bar{X}_2)] > 0 \qquad (2.3)$$

El lado izquierdo de la expresion (2.3) es llamado la funcion discriminante **lineal.**

Ejemplo: Notas en un curso

En una clase de Estadística Aplicada I consistente de 32 estudiantes, se toman la siguientes variables

E1: Nota del estudiante en el examen 1 (0-100)

E2: Nota del estudiante en el examen 2 (0-100)

PF: Promedio Final del estudiante en la clase (0-100)

Nota: Asume los valores P: si el estudiante pasa la clase y F si fracasa en la clase.

Los primeros 5 datos son:

	E1	E2	PF	Nota
1	96	100	100	p
2	96	94	99	p
3	100	91	97	p
4	93	96	97	p
5	90	94	95	p

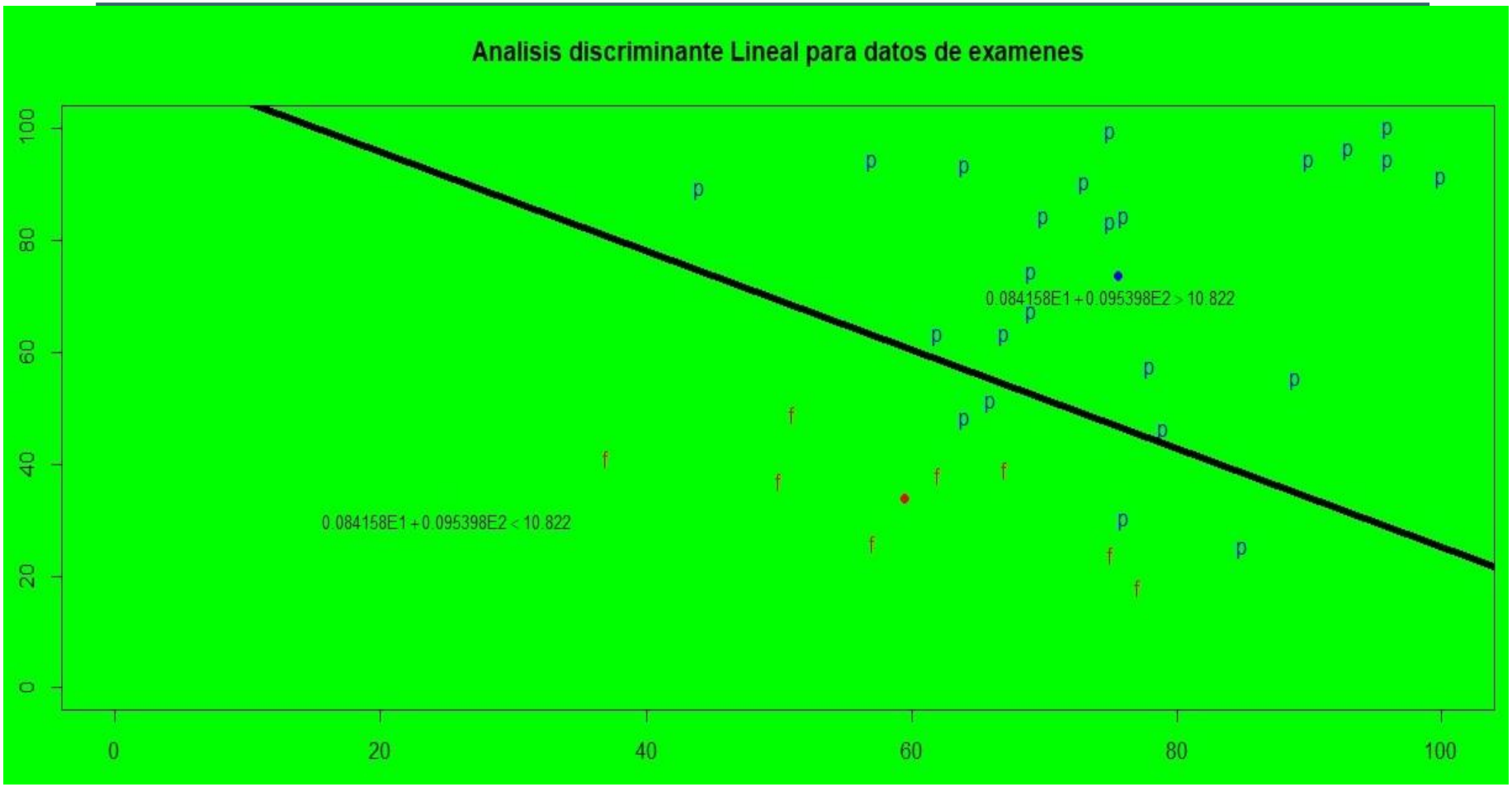
Se quiere predecir la Nota de un estudiante que tomará la clase en un próximo semestre, bajo las mismas condiciones actuales (mismo profesor, mismo texto, estudiantes de similar nivel en la clase, etc).

Ejemplo(cont)

```
#Separando el conjunto de datos en dos grupos
pasan=eje1dis[eje1dis[,3]=="p",]
npasan=dim(pasan)[1]
fracasan=eje1dis[eje1dis[,3]=="f",]
nfracasan=dim(fracasan)[1]
#Hallando las medias de E1 y E2 en ambos grupos
medp=apply(pasan[,1:2],2,mean)
> medp
      E1       E2
75.54167 73.75000
> medf=apply(fracasan[,1:2],2,mean)
> medf
  E1   E2
59.5 34.0
```

Ejemplo(cont)

```
#Hallando las matrices de covarianzas en ambos grupos
cov1<-cov(pasan[,1:2])
cov1
cov2<-cov(fracasan[,1:2])
cov2
 #Estimando la matriz de covarianza
> covcomb=((npasan-1)*cov1+(nfracasan-1)*cov2)/(npasan+nfracasan-
2)
> coeflda<-(medp-medf)%*%solve(covcomb)
> coeflda
            E1         E2
[1,] 0.08415817 0.09539823
> #Calculando el termino independiente
> indlda<-0.5*(medp-medf)%*%solve(covcomb)%*%(medp+medf)
> indlda
         [,1]
[1,] 10.82201
```


Analisis discriminante Lineal para datos de examenes
0.084158E1 + 0.095398E2 > 10.822
0.084158E1 + 0.095398E2 < 10.822
0
20
40
60
80
100

LDA(Fisher, 1936)

Fisher obtuvo la funcion discriminante lineal de la ecuacion (2.3) pero siguiendo otro camino. El trato de encontrar una combinacion lineal y de los atributos x's que separe las clases C1 y C2 lo mas posible asumiendo que hay homogeneidad de matrices de covariancias ($\Sigma 1=\Sigma 2=\Sigma$). Mas especificamente, si $y=\mathbf{d}'\mathbf{x}$, entonces, la distancia cuadrada entre las medias de y en cada clase dividida por la varianza de y en cada grupo esta dado por

$$\frac{(d'\mu_1 - d'\mu_2)^2}{d'\Sigma d} \qquad (2.4).$$

Esra razon es maximizada cuando $d=\Sigma^{-1}(\mu_1-\mu_2)$. Este resultado es obtenido aplicando la desigualdad de Cauchy-Schwartz (ver Rao, C. R. *Linear Statistical Inference and its applications*, p. 60).

LDA (cont)

- El numerador es llamado la suma of cuadrados entre grupos (BSS), y el denominador es llamado la suma of cuadrados dentro de grupos (WSS). Un estimado del valor **d** es $S^{-1}(\bar{x}_1 - \bar{x}_2)$.
- Fisher asigno un objecto **x a la** clase C1 si

$y=d'x=(\bar{x}_1 - \bar{x}_2)' S^{-1}$ **x** esta mas cerca a $\bar{y}_1 = (\bar{x}_1 - \bar{x}_2)'S^{-1}\bar{x}_1$ que a $\bar{y}_2$. El punto medio entre $\bar{y}_1$ y $\bar{y}_2$ es

$$\frac{(\bar{x}_1 - \bar{x}_2)'S^{-1}(\bar{x}_1 + \bar{x}_2)}{2}$$

- Notar que y es mas cerca a $\bar{y}_1$ si $y > \frac{\bar{y}_1 + \bar{y}_2}{2}$, esto da la ecuacion (2.3).

Analisis discriminante lineal como un clasificador Bayesiano

Supongamos que tenemos dos clases C_1 y C_2 que siguen una distribucion normal multivariada, Np($\mathbf{u}_1$,Σ_1) y Np($\mathbf{u}_2$,Σ_2) respectivamente y que además tienen igual matriz de covarianza $\Sigma_1=\Sigma_2=\Sigma$. Recordando que la distribucion Normal mutivariada esta dada por

$$\frac{1}{|\Sigma|^{1/2}(2\pi)^{p/2}}\exp(-1/2[(x-\mu)'\sum^{-1}(x-\mu)])$$

Luego la ecuacion (3.4) puede ser escrita como

$$\frac{\exp[-1/2(\mathbf{x}-\mathbf{u}_1)'\Sigma^{-1}(\mathbf{x}-\mathbf{u}_1)}{\exp[-1/2(\mathbf{x}-\mathbf{u}_2)'\Sigma^{-1}(\mathbf{x}-\mathbf{u}_2)} > \frac{\pi_2}{\pi_1} \qquad (3.5)$$

Tomando logaritmos en ambos lados se obtiene

$$-1/2[(\mathbf{x}-\mathbf{u}_1)'\Sigma^{-1}(\mathbf{x}-\mathbf{u}_1)-(\mathbf{x}-\mathbf{u}_2)'\Sigma^{-1}(x-\mathbf{u}_2)] > Ln(\frac{\pi_2}{\pi_1}) \qquad (3.6)$$

Despues de algunas simplificaciones resulta

$$(u_1-u_2)'\Sigma^{-1}(x-1/2(u_1+u_2))> Ln(\frac{\pi_2}{\pi_1}) \quad (3.7)$$

Si estimamos las medias poblacionales y la matriz de covarianza, se obtiene

$$(\overline{X}_1 - \overline{X}_2)'S^{-1}[x-(1/2)(\overline{X}_1+\overline{X}_2)] > Ln(\frac{\pi_2}{\pi_1})$$

esta desigualdad es similar a la que aparece en (2.3), excepto por el termino del lado derecho. Si ademas, consideramos que las probabilidades a priori son iguales ($\pi_1=\pi_2$) entonces si se obtendria la misma expresion (2.3).

Ejemplo(cont)

La funcion lda de R lleva a cabo el analisis discriminante lineal.
Rapidminer incluye el LDA . Debe elegirse primero el operador Modelling y luego el Classification and Regression y finalmente LDA .
En Python la libreria scikit-learn tiene LDA
From sklearn.discriminat_analysis import LinearDiscriminantAnalysis
Lda=LinearDiscriminantAnalysis()
Lda.fit(X,y)
En R:

```
> # LDA con priors iguales
> lda1<-lda(Nota~E1+E2,eje1dis,prior=c(.5,.5))
> lda1
>  plda1=predict(lda1,eje1dis[,-c(3,4)])$class
>   error1=sum(plda1!=eje1dis[,4])
>   error1
[1] 4
```

Ejemplo(cont)

```
>#matriz de confusion
> table(plda1,eje1dis$Class)
plda1  f  p
    f     8  4
    p     0 20
> #LDA con priors ponderadas
> lda2<-lda(Nota~E1+E2,eje1dis)
Prior probabilities of groups:
   f    p
0.25 0.75
>  plda2=predict(lda2,eje1dis[,-c(3,4)])$class
>   error2=sum(plda2!=eje1dis[,4])
>   error2
[1] 2
>Matriz de confusion
> table(plda2,eje1dis$Class)
plda2  f  p
    f  8  2
    p  0 22
```

La Matriz de Confusion

	Clase Verdadera=Si	Clase Verddera=No
Clase Predicha =Si	Positivos Verdaderos=TP	Positivos Falsos=FP
Clase Predicha=No	Negativos Falsos=FN	Negativos Verdaderos=TN

Exactitud=(TP+TN)/Total
Tasa de Mala Clasificacion=(FP+FN)/Total
=1- exactitud
Tasa de Positivos Verdaderos=Sensitividad=Recall=TP/(TP+FN)
Tasa de Falsos Postivos=FN(/FP+TN)
Especifidad=TN/(FP+TN)=1-Tasa de Falsos Negativos
Precision:TP/(TP+FP)
Prevalence=(TP+FN)/Total

LDA para mas de dos clases

Para G clases, el LDA asigna un objecto con vector de atributos x a la clase i tal que

$$i=\text{argmax}_k \ \bar{\mathbf{x}}'_{\mathbf{k}} \mathbf{S}^{-1}\mathbf{x} - (1/2)\bar{\mathbf{x}}'_{\mathbf{k}} \mathbf{S}^{-1}\bar{\mathbf{x}}_{\mathbf{k}} + Ln(\pi_k)$$

Para todo k en 1,2,…G. Al igual que antes el lado derecho es estimado usando la muestra de entrenamiento.

Ejemplo: El conjunto de datos Vehicle

```
Library(MASS)
ldaveh=lda(vehicle[,1:18],vehicle[,19])
predict(ldaveh)$posterior
predict(ldaveh)$class
# Estimating the error rate
mean(vehicle[,19]!=predict(ldaveh)$class)
[1] 0.2021277
```

Es estimado que 20.21% de las instancias estan mal clasificadas.

La tasa de error de mala clasificacion

Dado un clasificador d, su taza de error de mala clasificacion R(d) es la probabilidad de que d clasifique incorrectamente un objeto de una muestra (muestra de prueba) obtenida posteriormente a la muestra de entrenamiento.

Tambien es llamado el error verdadero.

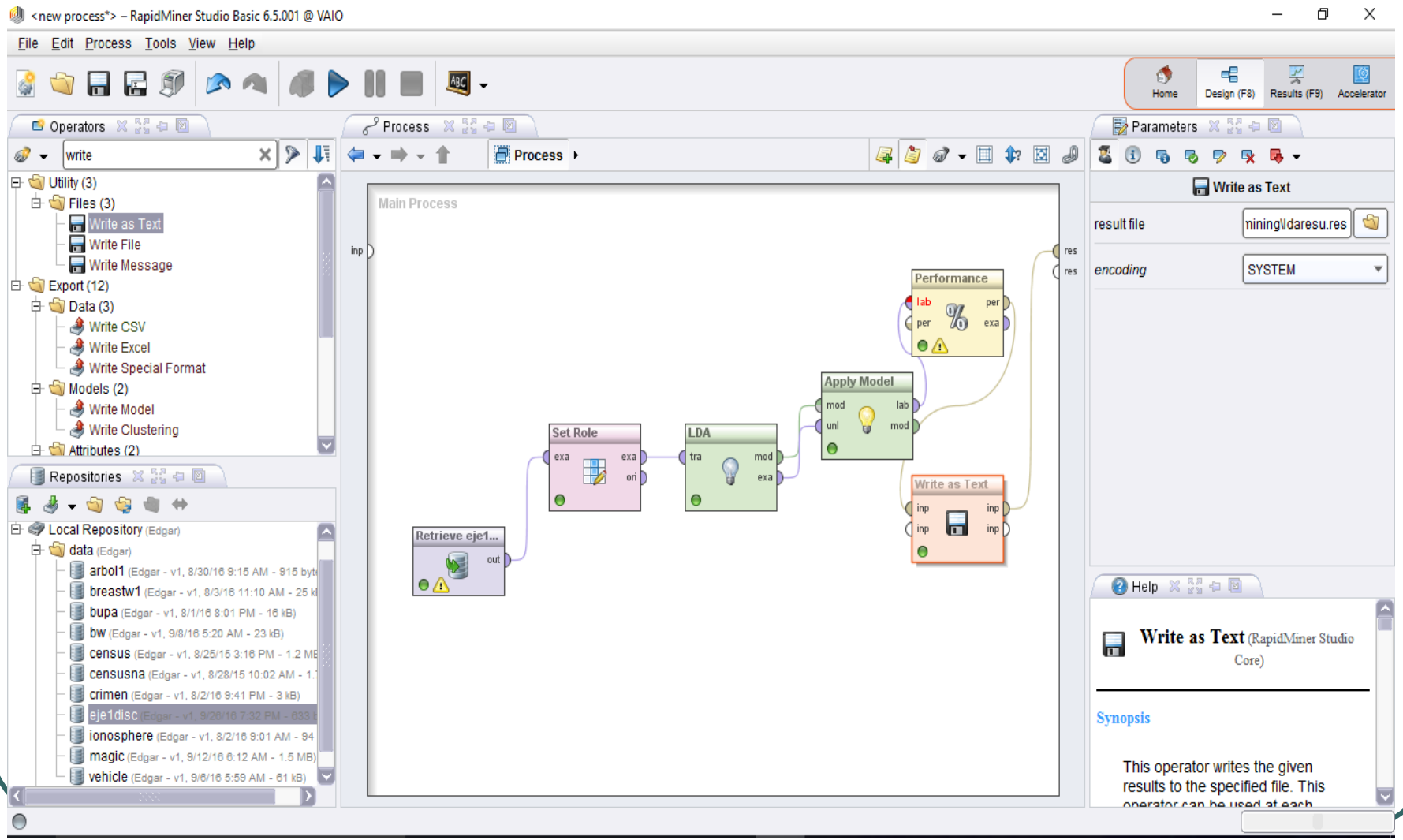
Es un valor desconocido que necesita ser estimado.

Metodos para estimar la taza de error de mala clasificacion

i) **Error por resubstitution or error aparente.** (Smith, 1947). Este es simplemente la proporcion de instancias en la muestra de entrenamiento que son incorrectamente clasificadas por la regla de clasificacion. En general es un estimador demasiado optimista y puede conducir a conclusiones erroneas cuando el numero de instancias no es demasidado grande comparado con el numero de atributos. Este estimador tiene un sesgo grande.

ii) **Estimacion dejando uno afuera (LOO). (**Lachenbruch, 1965). En este caso una instancia es omitida de la muestra de entrenamiento. Luego, el clasificador es construido y se obtiene la prediccion para la instancia omitida. Se registra si la instancia fue correcta o incorrectamente clasificada. El proceso es repetido para todas las instancias y la estimacion del error de mala clasificacion sera la proporcion de instancias clasificadas incorrectamente. Este estimador tiene poco sesgo pero su varianza tiende a ser grande.

Estimacion del error-Rapidminer



Examples of LOO

```
> ldaexa=lda(eje1dis[,1:2],eje1dis[,4],CV=TRUE)
> mean(eje1dis[,4]!=ldaexa$class)
[1] 0.0625
> ldaveh=lda(vehicle[,1:18],vehicle[,19],CV=TRUE)
> mean(vehicle[,19]!=ldaveh$class)
[1] 0.2210402
```

Metodos para estimar la tasa de error de mala clasificacion

iii) Validacion cruzada. (Stone, 1974) En este caso la muestra de entrenamiento es dividida al azar en v partes (v=10 es lo mas usado). Luego, el clasificador es construido usando todas las partes menos una. La parte omitida es considerada como la muestra de prueba y se hallan las predicciones de cada una de sus instancias. La tasa de error de mala clasificacion CV es hallada sumando el numero de malas classificaciones en cada parte y dividiendo el total por el numero de instancias en la muestra de entrenamiento. El estimado CV tiene poco sesgo pero una alta variabilidad. Para reducir dicha variabilidad se repite la estimacion varias veces.

Ejemplo: Usando la libreria dprep

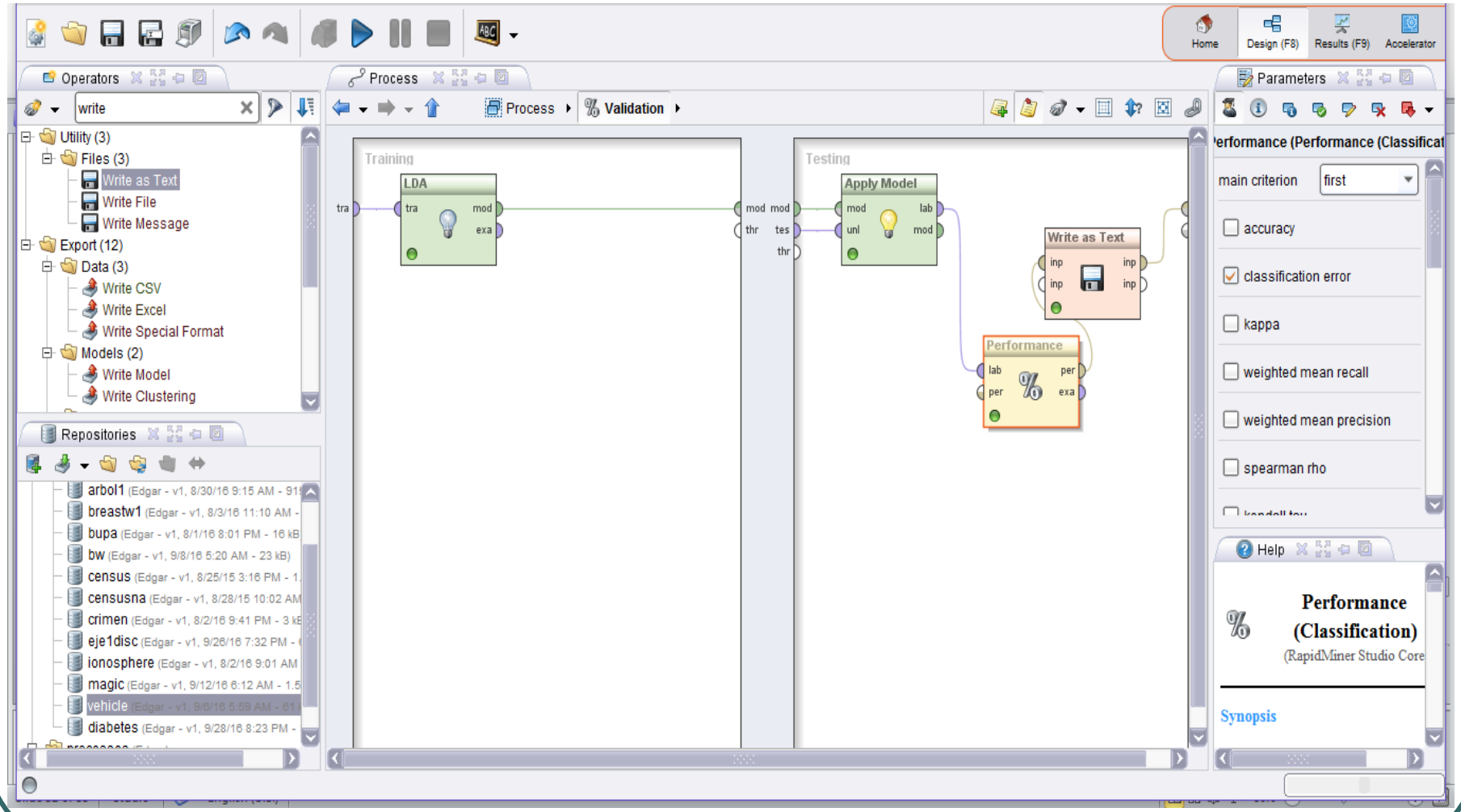
crossval(eje1dis,method="lda",repet=20)
[1] 0.096875
crossval(vehicle,method="lda",repet=10)
[1] 0.2206856

iv) El metodo de retencion. Aqui se extrae al azar un porcentaje de los datos (70%) y es considerado como la muestra de entrenamiento y los restantes datos son considerados como la muestra de prueba. El clasificador es evaluado en la muestra de prueba. El experimento es repetido varias veces y luego se promedia la tasa de error en las muestras de prueba.

v) Bootstrapping. (Efron, 1983). En este metodo se generan de la muestra de entrenamiento varias muestras con reemplazo. Las que luego se usan para construir el clasificador. La idea es reducir el sesgo del error aparente. Es casi insesgado pero tiene una varianza grande. Su costo computational es alto.

Existen varias variantes de este metodo.

Cross-validation in Rapidminer



Un metodo consiste en estimar el sesgo del error por resubstitución (eA) de la muestra de entrenamiento siguiendo los siguientes pasos:
1) Hallar el clasificador usando la muestra original y su error por resubstitución (eA)
2) Generar una muestra con reemplazo de la muestra de entrenamiento, que tenga su mismo tamaño esta es llamada la muestra "boostrapeada".
3) Hallar el clasificador usando la muestra "bootstrapeada" y obtener el error por resubstitucion para la muestra "bootstrapeada" (eb)
4) Considerar la muestra "bootstrapeada" como la muestra de entrenamiento y la muestra original como la muestra de prueba y hallar el error del clasificador cuando es aplicada a esta última (et). Este es un estimado del error verdadero o actual.
5) Calcular et-eb
6) Repetir los pasos 2-5 B times (B entre 50 y 100)
7) El promedio de las diferencias obtenidas en el paso 5 (Wboot) será el estimado del sesgo
8) El estimado de la tasa de clasificación errada sera eA+Wboot .
Existen tambien los métodos de doble "bootstrapping" y el estimator .632