PUCE Virtual
EDUCACIÓN FLEXIBLE

Estadística 1 para administración

Métricas para medir bondad de modelos de clasificación

Clase 14

La excelencia no se improvisa

Introducción de la clase

En la presente clase revisaremos más en profundidad los modelos logísticos a fin de entender las transformaciones que se deben realizar, así como las distintas métricas que sirven para evaluar los modelos de clasificación, como veremos, existen varias métricas y su uso y aplicabilidad va a depender de algunos factores y de el propósito general de clasificación.
Un tema de particular importancia será el tratamiento de la data cuando las clases se hallan desbalanceadas, explicaremos los resultados de la métrica cuando esto ocurre, así como los mecanismos para compensar el desbalanceo.

Resultado o resultados de aprendizaje que será abordado con el contenido de la clase
Presentar informes estadísticos interpretando los resultados dentro del contexto del problema.

Clase 14: Métricas para medir bondad de modelos de clasificación

Antes de entrar en el análisis de las métricas para evaluar la bondad de los modelos es importante saber interpretar los coeficientes de las variables en este tipo de modelos y analizar como llegamos a ellos.

Recordando la definición de logit vista en módulo anterior.

$$logit(P) = \log\left(\frac{P}{1-P}\right)$$

Recordando la definición del Odds

$$Odds = \frac{P}{1-P}$$

El Odds ratio que nos permite comparar los Odds en dos grupos

$$OddsRatio = Odds_{grupo1} / Odds_{grupo2}$$

Como estamos utilizando modelos GLM queremos que el valor de la respuesta sea igual a la componente sistemática del modelo GLM

$$\beta_0 + \beta_1 X_1$$

Sin embargo, en la función logit el eje X tiene como valores posibles un rango entre 0 y 1

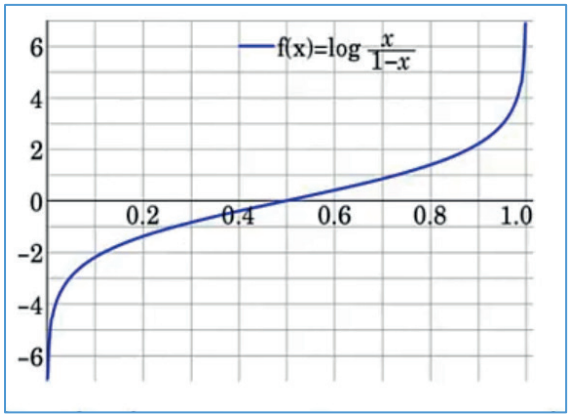


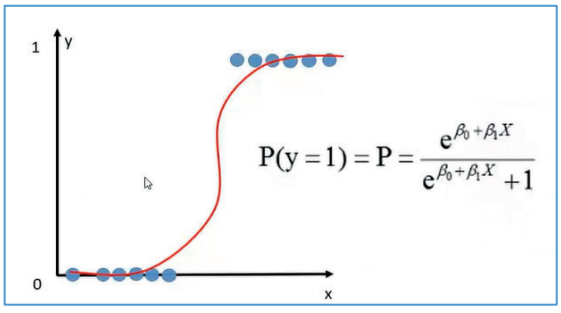
Figura 1
Característica la función logit
Tomado de Jurafsky D. Martin J. (2024)

Por lo tanto, tenemos que invertir a la función logit para que este sea nuestro valor de salida

$$logit^{-1}(x) = \frac{e^x}{1 + e^x}$$

Lo cual nos da como resultado la siguiente curva característica conocida como sigmoidea.

Figura 2
Característica la función logit invertida
Tomado de Jurafsky D. Martin J. (2024)



La función sigmoidea nos permite tomar una instancia x y calcular la probabilidad P(y = 1|x). ¿Cómo decidimos qué clase aplicar a una instancia de prueba x? Para una x dada, decimos que sí si la probabilidad P(y = 1|x) es mayor que 0,5, y no en caso contrario. Llamamos 0,5 al límite de decisión:

Sin embargo, como el modelo debe ser una función lineal se debe aplicar una transformación de tal forma que sea igual al componente sistemático de GLM, lo cual lo logramos reduciendo la fórmula sigmoidea y aplicando logaritmo a cada lado y obtendríamos:

$$\log\left(\frac{P}{1-P}\right) = \beta_0 + \beta_1 X_1$$

Entonces la función exp(β_0) representa las probabilidades de que la característica (variable) de éxito esté presente para un individuo x=0, es decir en la línea de base. Si hay múltiples predictores involucrados, todos deberían establecerse en 0 para esta interpretación.

La función exp(β_i) representa el aumento multiplicativo en las probabilidades de éxito por cada aumento de 1 unidad en x. Si $\beta_i > 0$ indica que es una relación positiva, si es negativo entonces el incremento de X decrementa la posibilidad de que se presente el evento.

Por ejemplo, de acuerdo con el modelo creado en la clase anterior el coeficiente de la variable "MaritalStatus.Single" es 0.8421 , y el exp(0.8421) es 2.32. Esto debe interpretarse de la siguiente forma: Por cada unidad que aumenta la variable MaritalStatus.Single el **odds** (no el OR) de que se presente la renuncia aumenta en 2.32. Note que en este modelo se ha aplicado la codificación One-Hot previamente, por esto existen 3 variables "dummy" correspondiente a los 3 niveles del factor. Por otro lado, si la fórmula hubiera sido Attrition~MaritalStatus, glm habría implementado automáticamente la codificación mediante variables dummy, esta es similar a one-hot excepto que se habrían creado solo 2 variables dummy, más una categoría de referencia, cuando todas las demás variables dummy sean 0 esa categoría se usará como valor base del modelo.

Figura 3
Modelo creado en la clase anterior
Creación de autor Alfonso Prado

```
Call:  glm(formula = Attrition ~ MaritalStatus.Divorced + MaritalStatus.Single +
    MaritalStatus.Married, family = binomial(link = "logit"),
    data = data_dv)

Coefficients:
           (Intercept)  MaritalStatus.Divorced    MaritalStatus.Single
               -1.9235                 -0.2843                   0.8421
 MaritalStatus.Married
                    NA

Degrees of Freedom: 4299 Total (i.e. Null);  4297 Residual
Null Deviance:      3804
Residual Deviance: 3680         AIC: 3686
```

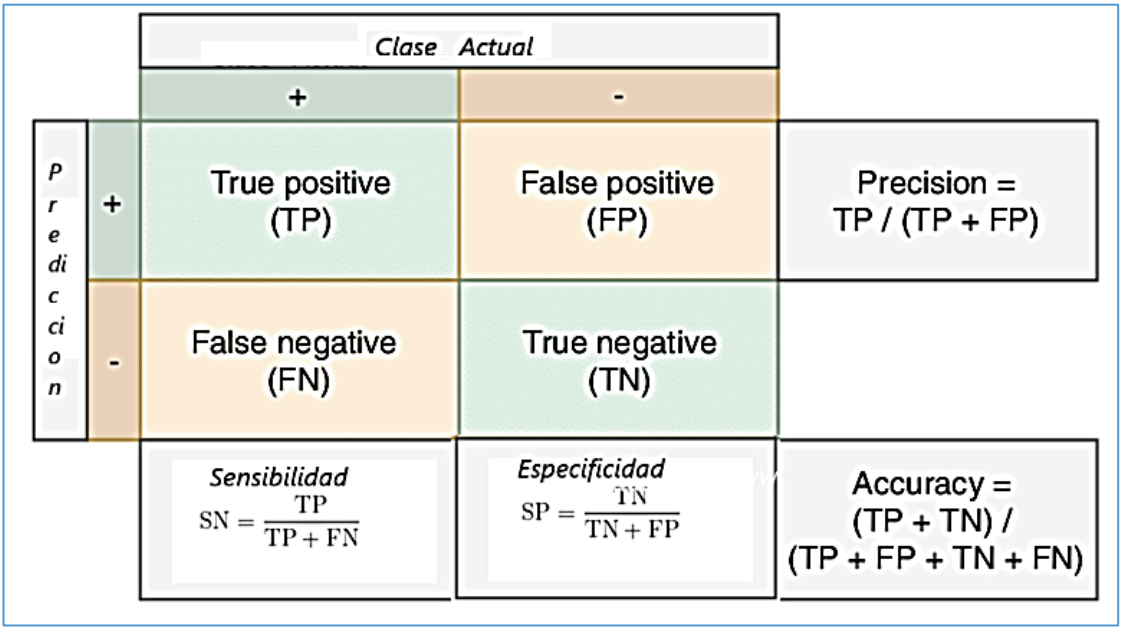
Matriz de confusión

Es importante mencionar que para optimizar el rendimiento de cualquier modelo debe primero realizarse las actividades de limpieza de datos, pre procesamiento que incluye la gestión de datos atípicos, transformaciones, normalización entre otros.

Luego nos preguntamos ¿cómo podemos medir la efectividad de nuestro modelo en términos de eficiencia, rendimiento y precisión? Ahí entra la matriz Confusión.
De acuerdo a Hilbe (2015), esta matriz es la medida del rendimiento para la clasificación y es ampliamente utilizada en el aprendizaje automático (ML).

La matriz de confusión es un tipo especial de tabla de contingencia, de dos dimensiones ("real" y "predicha") y conjuntos idénticos de "clases" en ambas dimensiones (cada combinación de dimensión y clase es una variable en la tabla de confusión). El nombre proviene del hecho de que facilita ver si el sistema está confundiendo dos clases (es decir, etiquetando erróneamente una como otra). Generalmente, las tablas contienen la predicción en las filas y los valores actuales (observaciones) en las columnas, aunque hay autores que lo colocan al revés.

Tabla 1
Tabla de confusión métricas y fórmulas, Prado (2025)
Adaptación de la tabla https://www.doptsw.com/posts/post_2024-10-02_c1345



En la tabla vamos a identificar entonces 4 valores:

- Verdaderos positivos: Identificados como TP predicción positiva y en realidad es positivo.
- Falsos positivos: Identificados como FP predicción positiva y en realidad es negativo
- Falsos negativos: Identificados como FN predicción negativa y en realidad es positivo.
- Verdaderos negativos: Identificados como TN predicción negativo y en realidad es negativo

Métricas de la tabla de confusión

Basados en la matriz de confusión podemos establecer algunos indicadores de la eficiencia del modelo:

Accuracy: Simplemente mide con qué frecuencia el clasificador hace la predicción correcta. Es la relación entre el número de predicciones correctas y el número total de predicciones, se utiliza cuando los verdaderos positivos y los verdaderos negativos son más importantes.

Sensibilidad: Calculada como el número de predicciones positivas correctas dividido por el número total de positivos reales (observaciones). Algunos autores la denominan recuperación (REC) o tasa de verdaderos positivos (TPR).

Especificidad: Calculada como el número de predicciones negativas correctas dividido por el número total de negativos. Algunos autores la denominan tasa negativa verdadera (TNR).

Tasa de Falsos Positivos (FPR): Se define como la relación entre falsos positivos y la suma de falsos positivos más verdaderos negativos. Es el inverso de la especificidad (1-especificidad).

Precisión: Representa la proporción de casos positivos que se predicen correctamente dividido para el total de casos positivos que se predicen. La diferencia entre sensibilidad y precisión radica en que la sensibilidad se calcula en base a los casos reales (observaciones) que son positivos, mientras que la precisión se calcula en base a las predicciones positivas.

La pregunta que nos hacemos es ¿Cuál de las métricas deberíamos utilizar? Todos los indicadores anteriores sirven para verificar una parte del modelo, por ejemplo, que tan bien se predice los TP o TN y funcionan bien cuando la data esta balanceada. Se considera que la data es balanceada cuando la cantidad de casos de cada clase es similar y desbalanceada cuando existen diferencias significativas.

Veamos un ejemplo: Asuma que tenemos 1000 casos positivos de los cuales el 90% son predichos correctamente y 10% incorrectamente. Asuma que tenemos 50 casos negativos de los cuales podemos predecir correctamente el 50%. De la Tabla 1 obtenemos: TPR=900/925 =0.97 =97% y TNR=25/125=0.2 =20% . Notamos que TPR (REC) está sobreestimado y TNR esta subestimado, este es el efecto que se obtiene cuando la data no está balanceada.

En estos casos debemos recurrir a la métrica llamada F1 o a las curvas características que resumen de mejor manera la eficiencia del modelo.

Figura 4

Fórmula de la métrica F1

Adaptación de la fórmula de Hilbe J., (2015)

$$PuntajeF1 = \frac{2 * Precision * Sensibilidad}{Precision + Sensibilidad}$$

Otras alternativas posibles en caso de data desbalanceada consisten en previamente balancear la misma en forma manual, para luego entrenar el modelo. Existen dos técnicas para balancear conocidas como: Undersampling y Oversampling:

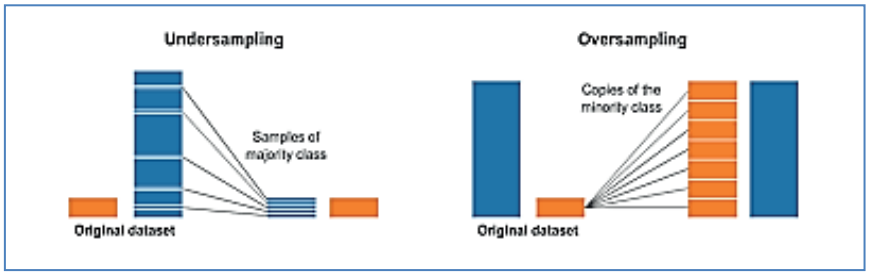
- ✓ El "Under-sampling" (sub muestreo) funciona reduciendo el tamaño de la clase mayoritaria para que coincida con la clase menos prevalente.
- ✓ El "over-sampling" (sobre muestreo) se utiliza para aumentar el tamaño de muestras minoritarias. La técnica conocida como SMOTE (Synthetic Minority Oversampling Technique) ofrece esta posibilidad en el paquete caret.

La siguiente figura explica el concepto

Figura 5

Undersampling (sub muestreo) y Oversampling (sobre muestreo)

Fuente: Roweida M. Et. Al. (2020)



La implementación del sub muestreo es relativamente fácil, implica eliminar una cierta cantidad de observaciones, pero en sobre muestreo se podría pensar en duplicar las observaciones de dicha clase lo cual equilibrará la distribución de clases, pero esto no mejorará el rendimiento del modelo, ya que no le proporciona información adicional.

SMOTE

Esta técnica selecciona los ejemplos minoritarios que están cerca del espacio de características. En ML las "características" son sinónimo de variables y por lo tanto el espacio de características son el espacio de valores de dicha variable. Dentro de este espacio, SMOTE crea una nueva muestra en algún lugar y le asigna el valor del vecino más cercano utilizando el algoritmo de KNN (K Nearest Neighbors). En otras palabras, el algoritmo crea un ejemplo aleatorio cuya clase se fija de acuerdo con el K vecino más cercano. Esta observación "sintética" se crea entre dos ejemplos en el espacio de características.

El uso de SMOTE tiene una desventaja, ya que no considera la clase mayoritaria al crear ejemplos sintéticos. En otras palabras, el algoritmo funciona bien cuando las clases están bien separadas, esto puede causar problemas cuando hay una superposición entre las clases.

SMOTE

Presenta detalle para su implementación

Enlace

Veamos un ejemplo con SMOTE

```
#Validamos si la data esta balanceada
prop.table(table(data$Attrition))
```

```
        No        Yes
0.8383721 0.1616279
```

```
#Obviamente no está balanceada
#En este caso utilizaremos la función downSample cuya sintaxis es:
#downSample(x, y, list = FALSE, yname = "Class")
#x=lista de variables predictoras
#y=variable de clase, en este caso Attrition
#list=FALSE indica que es un dataframe
#yname= nombre de la etiqueta de clase


set.seed(100)
dataBalanceada <- caret::downSample(x=data %>% select(-Attrition) ,
```

```
                    y=data$Attrition,
                    yname="Attrition")


prop.table(table(dataBalanceada$Attrition))
  No Yes
0.5 0.5

#Ahora deberíamos repetir la creación y análisis de la matriz de confusión para asegurar que sensibilidad y especificidad están correctos
```

En última instancia, la mejor métrica es la que le conviene al negocio. Por ejemplo, asuma que el cliente está usando un modelo de clasificación para detectar transacciones fraudulentas. En este caso el cliente estará más interesado en calcular la sensibilidad, en otros negocios el cliente puede enfocarse más bien en los casos negativos y en este caso utilizará la especificidad.

Curvas Características de la Separabilidad

Estas curvas permiten visualizar la separabilidad entre clases, idealmente, se desea que las observaciones positivas sean predichas como positivas y lo mismo con las negativas. Pero en la práctica los modelos nunca son perfectos.
Estas curvas ayudan a visualizar la bondad de un modelo de clasificación.

Curva ROC

Una curva ROC es un gráfico que muestra el rendimiento de un modelo de clasificación en todos los umbrales de clasificación. Esta curva usa dos métricas en sus ejes, la tasa de verdaderos positivos TPR (sensibilidad) y la tasa de falsos positivos FPR (1-especificidad).

En principio desearíamos que la sensibilidad sea lo más grande posible, mientras el FPR sea lo más bajo posible. La gráfica resultante sería la figura 7(a)

En la medida que el modelo empieza a perder su capacidad de discriminar, entre clases positiva y negativa, la curva empieza a acercarse a una diagonal. Cuando llega a ser una diagonal es la peor situación porque indica que el modelo no puede discriminar entre clases y por lo tanto no es utilizable.

Cuando la tasa de falsos positivos llega a ser igual a1 el modelo en realidad está invirtiendo las clases. Significaría que el modelo predice una clase negativa como una clase positiva y viceversa, en cuyo caso la curva ROC se mira como en la figura 7(c). Dado que TPR y FPR tiene un valor máximo de 1 , el área bajo la curva tendrá un valor de 1.

De acuerdo a Hilbe (2015) “Los valores de 0,5 a 0,65 tienen un poder predictivo bajo. Los valores de 0,65 a 0,80 tienen un valor predictivo moderado. Muchos modelos logísticos se ajustan a este rango. Los valores superiores a 0,8 e inferiores a 0,9 generalmente se consideran de alto poder predictivo”

La curva se crea basado en un umbral del valor de la variable, para cada umbral escogido se calcula las dos métricas mencionadas y se gráfica el punto que luego se une por medio de una curva, dando como resultado la siguiente figura.

Figura 6

Casos extremos de la curva ROC

Creación del autor Alfonso Prado

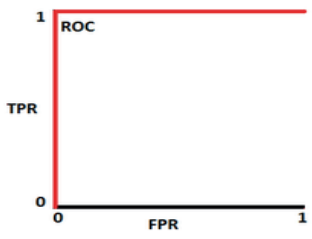
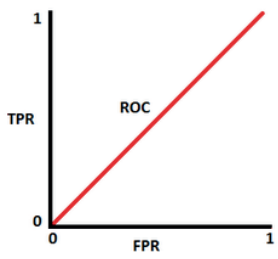
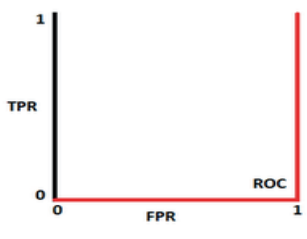
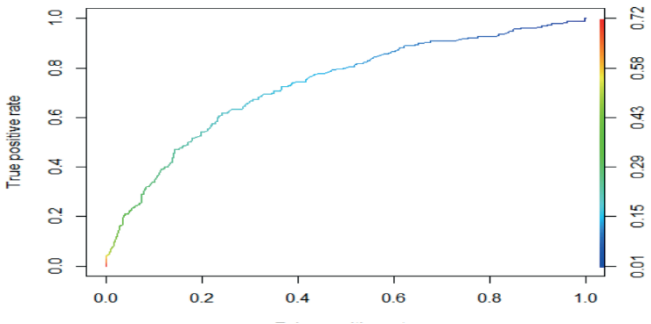
Modelo ideal(a)	Modelo sin capacidad de discriminación(b)	Modelo invierte las clases(c)
		

Figura 7

Curva ROC de un modelo real

Creación del autor Alfonso Prado



A fin de poder comparar modelos se estila calcular el área bajo la curva, de esta manera pequeñas diferencias entre curvas de distintos modelos son fáciles de notar. Un modelo excelente tiene AUC cerca de 1, lo que significa que tiene una buena medida de separabilidad entre las clases. Un modelo pobre tiene un AUC cercano a 0.5 lo que significa que tiene la peor medida de separabilidad, esto es, no puede distinguir entre una clase y otra, y no es mejor que clasificar al azar. Medidas menores a 0.5 indicaría que los casos positivos se predicen como negativos y viceversa.

Curva PR sensibilidad vs precisión

La curva PR es la gráfica resultante de relacionar la precisión y la sensibilidad (REC o TPR) de un modelo. Nos permite ver a partir de qué valor de sensibilidad obtenemos una degradación de la precisión.

Note que en realidad lo que está comparando es FP y FN, es decir la diagonal secundaria. En un modelo perfecto sería aquel que pase por el punto (1,1), por tanto, cuanto más se acerque a esa esquina superior derecha mejor es el comportamiento del modelo.
Al igual que con la curva ROC, también se puede calcular el área bajo la curva.

Es preferible usar esta curva en vez de la curva ROC en casos en los que exista un desbalanceo en las clases. Esto se debe a que la curva ROC puede dar una visión optimista del modelo por la dependencia de los falsos negativos que en conjuntos con clases desbalanceadas aumentará.

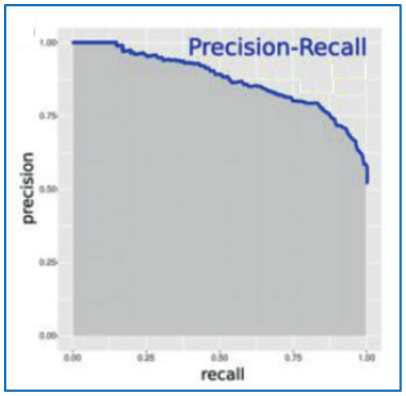


Figura 8

Curva precisión vs sensibilidad

Creación del Autor Alfonso Prado

Referencias citadas en la Clase 14

Roweida M. Et. Al. (2020) , *Machine Learning with Oversampling and Undersampling Techniques*, Jordan University of Science and Technology https://www.researchgate.net/publication/340978368_Machine_Learning_with_Oversampling_and_Undersampling_Techniques_Overview_Study_and_Experimental_Results/download?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6Il9kaXJlY3QiLCJwYWdlIjoiX2RpcmVjdCJ9fQ

Jurafsky D. , . Martin J. (2024) , *Speech and Language Processing*. https://web.stanford.edu/~jurafsky/slp3/

Hilbe J., (2015) , Practical Guide to Logistic Regression, CRC Press

Agresti, A., (2007), An introduction to categorical data analysis, John Wiley and Sons publication

Webster A., (2000), Estadística aplicada a los negocios, Irwin Professional Publishing

Schmuller J.(2022), Statistical Analysis with Excel For Dummies, Wiley

Teck K, (2022) , Practical t-test Power Analysis with R, Practical Assessment, Research & Evaluation , Volumen 27, No 18

Berman H.G., "*F Distribution* , descargado de https://stattrek.com/probability-distributions/f-distribution

Rodrigo J. (2016), *Análisis de varianza para comparar múltiples medias*, https://cienciadedatos.net/documentos/19_anova

Szretter, M. (2017), Apunte de Regresión Lineal, https://mate.dm.uba.ar/~meszre/apunte_regresion_lineal_szretter.pdf

Estadística Inferencial https://www.studocu.com/cl/document/universidad-de-valparaiso/estadistica-inferencia

PUCE Virtual

La excelencia no se improvisa

síguenos

CV_ICYS_M