Diseño experimental en analítica de datos

PID_00247924

Ester Bernadó

Tiempo mínimo de dedicación recomendado: 3 horas

Universitat Oberta de Catalunya

Índice

Introducción

Hace un tiempo que Pedro González trabaja como consultor de analítica de datos. Sus clientes le piden a menudo que analice datos de sus negocios y que les proporcione la mejor solución a sus problemas de analítica. Después de unos años de experiencia se da cuenta de que todos sus clientes se encuentran, en el fondo, con problemas similares. Incluso hace unos días el programador con el que trabaja le preguntó: «¿Cuál es el mejor método de clasificación? Me han hablado de los árboles de decisión. Si implemento una versión muy optimizada, nos ahorramos futuras implementaciones de otros algoritmos». Otro día, un cliente, la sra. Martínez, directora de la sección de analítica del departamento de marketing de una empresa, comentaba: «Creo que las máquinas de soporte vectorial son muy buenos métodos de clasificación. A partir de ahora, las aplicaré en todos los problemas de analítica de marketing». Otros clientes preguntan por el porcentaje de acierto de los clasificadores o el error medio de las predicciones. Por ejemplo, el sr. Dupont preguntó si el sistema de predicción de ventas de su ecommerce era fiable.

Con los años de experiencia, Pedro se ha dado cuenta de algunas verdades:

- *No hay un método infalible que tenga el mejor rendimiento para todos los problemas de analítica. A pesar de ello, sus clientes continúan pidiendo la «receta mágica». Pedro responde que no existe el* one-size fits all.

- *Para un problema dado, es necesario realizar una estimación de la precisión o error del modelo y comparar distintos modelos entre sí para identificar la mejor aproximación al problema.*

- *La estimación de la calidad del modelo debe hacerse con cariño, refiere Pedro. Es decir, hay que evitar sesgos, ya sean estos pesimistas u optimistas. Dice que es difícil justificar a un cliente que su estimación de precisión era del 90 % y finalmente, el modelo opera con precisiones del 50 %.*

Pedro ha firmado un proyecto recientemente para el desarrollo de un sistema de predicción de fraude en transacciones con tarjetas de crédito. El cliente dispone de una base de datos de transacciones a partir de la cual se puede construir un modelo. Pedro, con ayuda del informático, desarrollará el sistema de predicción de fraude. Además, deben proporcionar una estimación del porcentaje de acierto del sistema. Es decir, cuando se prediga una transacción como fraudulenta o no fraudulenta, qué probabilidad tiene de acertar en su predicción. En la primera fase, se debe realizar el diseño experimental del proyecto.

Como se puede observar en la historia que inicia este módulo, el diseño experimental del análisis de datos es de vital importancia. En un proyecto de analítica no solo es necesario descubrir patrones interesantes en los datos, también

es importante poder estimar qué calidad presenta el modelo extraído o los patrones identificados. Esta calidad puede estar medida de distintas formas, siendo la precisión o el error algunas de las más habituales. En este módulo se revisan los elementos esenciales del diseño experimental de análisis de datos, las medidas de calidad de los modelos y la validación estadística de los resultados.

1. Diseño experimental de análisis de datos

Existen varias metodologías que especifican los pasos necesarios para realizar un análisis de datos. Una de las más conocidas es CRISP-DM (*Cross Industry Standard Process for Data Mining*), propuesta por Daimler-Chrysler y SPSS, que fueron pioneros en el uso de la minería de datos en sus procesos de negocio. Otra metodología conocida es SEMMA (*Sample, Explore, Modify, Model, Assess*), propuesta por el SAS Institute y definida como el proceso de selección, exploración y modelado de grandes cantidades de datos para descubrir patrones desconocidos. Otras metodologías son las propuestas por Zumel y Mount (2014) y Jain y Sharma (2015). Estos últimos introducen el *framework* BADIR (*Business question, Analysis plan, Data collection, Insights, Recommendation*). Aunque cada metodología tiene algunas particularidades, todas tienen muchos aspectos en común, puesto que recogen los pasos básicos que un analista debe realizar para desarrollar proyectos de analítica. Todas ellas presentan un conjunto de fases que se mueven desde la definición del problema, contextualizado en el entorno de negocio o científico, hasta la implantación y toma de decisiones. Los procesos suelen ser interactivos, en diálogo continuo con el analista y agentes clave (*stakeholders*) e iterativos, ya que progresan en iteraciones sucesivas, y con posibles realimentaciones, donde el resultado de una fase puede provocar el replanteamiento de fases previas.

Para el propósito de este módulo, tomaremos como referencia el proceso de analítica de datos propuesto por Zumel y Mount, en el que se enumeran las siguientes fases:

1) Definir el objetivo
2) Recolectar y gestionar los datos
3) Construir el modelo
4) Evaluar críticamente el modelo
5) Presentar los resultados y documentarlos
6) Implantar el modelo

En la primera fase, se plantean los **objetivos del análisis**. Estos pueden estar contextualizados en un entorno científico, con intereses de investigación, o un entorno aplicado o de toma de decisiones como por ejemplo en el ámbito de un negocio. En cualquier caso, hay una necesidad de análisis de datos que debe formularse de forma explícita y que marcará la definición del resto de las fases.

En la **recolección** de datos se identifican los datos de interés y se obtienen estos datos. La fuente de datos puede provenir de datos previamente almacenados o bien se diseña un proceso de recolección de datos *ex profeso*. En esta fase se debe tener en cuenta el muestreo de los datos para conseguir muestras re-

presentativas del objeto de estudio, puesto que las conclusiones que se extraigan de los datos dependen en gran medida de esta representatividad. Las técnicas de muestreo descritas en el módulo «Introducción a la Estadística» (como el muestreo probabilístico, y dentro de este, el muestreo estratificado simple) promueven esta representatividad tan necesaria. Cuando se dispone de la muestra, es frecuente realizar un preproceso sobre la misma, especialmente si esta proviene de repositorios de datos sobre los que el analista no ha tenido posibilidad de intervención. Así pues, se revisan los datos y se procede a la limpieza de datos, como detección de errores en los mismos, tratamiento de valores perdidos y transformación de datos para su modelado posterior.

Una vez se dispone de la muestra preparada para el análisis, se **construye el modelo** de los datos. Esta es la fase central del análisis donde se extraen los patrones de los datos (muchas veces denominados *insights*). En esta fase se debe decidir qué modelos son más apropiados para extraer la información en función del objetivo del análisis y los recursos disponibles. Esta fase se denomina *fase de entrenamiento* del sistema (*training*), que se distingue de la *fase de test* que viene a continuación.

La fase 4 de la metodología consiste en **evaluar críticamente el modelo**, lo cual significa que se deben proporcionar medidas de la calidad del modelo o, lo que es lo mismo, de la calidad de la información extraída de los datos. En este paso no nos referimos todavía a si los *insights* son útiles para el contexto, lo cual se debe analizar en un momento posterior, sino más bien con qué precisión o confianza podemos afirmar estos *insights*. Por ejemplo, si un *insight* extraído es que los clientes que compran en el supermercado los lunes se gastan un 50 % menos que los que compran en sábado, deberíamos poder acompañar esta afirmación con la precisión de la misma. Podríamos decir que esto se cumple en un 85 % de los casos. Si se obtienen modelos que no son suficientemente precisos, se pueden replantear algunas de las fases anteriores: ya sea el método usado para extraer los *insights*, su configuración, la representatividad de la muestra o el propio enfoque del análisis.

Asimismo, en esta fase es interesante situar la información extraída dentro del contexto de definición del problema. Se deben revisar cuestiones como si la información extraída es útil para los objetivos de la investigación o del ámbito de decisión. Puede que la información sea precisa (por ejemplo, el comportamiento extraído por los clientes del supermercado podría tener una precisión del 95 %) pero no aportar nada nuevo en relación a los objetivos de análisis. Si es así, se deben revisar las fases previas para abordar otros enfoques.

En las fases 5 y 6 se **presentan los resultados** ante los agentes clave y se **desarrolla e implanta** el sistema o se toman las decisiones oportunas. Como corresponde a las fases finales de un proceso, también se revisa si es necesario realizar nuevas etapas en el proyecto que arrojen nueva información con más datos o mediante nuevos enfoques.

El aspecto central de este módulo y que configura gran parte del proceso de analítica de datos es la fase 4, la evaluación del modelo. Partimos de la suposición de que el analista ya ha realizado el muestreo y limpieza de datos y se dispone a construir el modelo. Antes de construirlo, se necesita diseñar cómo se va a construir el modelo, es decir, bajo qué condiciones y cómo se evaluará la calidad del mismo. Precisamente, se hace necesario muestrear de nuevo los datos para seleccionar cuáles de ellos formarán parte de la construcción del modelo y cuáles se reservarán para contrastar la calidad del modelo. Es necesario también configurar con qué métricas se evaluará el modelo, de forma que se puedan recoger estas medidas tanto durante la construcción del modelo como al finalizar el proceso. Es por todo ello que decimos que la evaluación de la calidad del modelo determina en gran medida el proceso de analítica de datos. En el apartado siguiente se introducen brevemente las distintas métricas de calidad de los métodos principales del análisis de datos. A continuación, se describen los procesos de muestreo necesarios vinculados a la estimación de la calidad del modelo. Finalmente, se trata la validación estadística que permite medir en qué grado las conclusiones obtenidas del análisis son válidas estadísticamente.

2. Métricas

En esta sección se presentan las medidas habituales de evaluación de la calidad de los modelos. El tipo de métrica depende del modelo que se extrae de los datos y este, a la vez, depende de la pregunta de investigación o necesidad de información planteada en los objetivos del proyecto de analítica. Así, si la pregunta es qué tipologías de clientes compran en un supermercado, el modelo será de agrupación o *clustering* y su evaluación tendrá que medir en qué grado los grupos identificados cumplen determinados criterios deseables para una buena agrupación. Si el objetivo del proyecto es detectar vertidos de petróleo en el océano, el modelo construido es de clasificación o categorización (una mancha sospechosa debe clasificarse como vertido o no) y la evaluación del mismo es el grado de precisión, o dicho de otra forma, la capacidad de detección de estos vertidos.

En la tabla 1 se resumen las métricas correspondientes a los modelos más frecuentes de análisis de datos. Concretamente, nos centramos en los modelos de clasificación, regresión, asociación y agrupación, los cuales se describen a continuación.

Tabla 1. Modelos y métricas de calidad de análisis de datos

Ejemplo	Modelo	Métricas
Detección de vertidos de petróleo a partir de imágenes de satélite	Clasificación	• Precisión (*accuracy*) • Kappa • Sensitividad • Especificidad
Predicción del área quemada de bosque	Regresión	• Error absoluto medio • Error cuadrático medio • Raíz del error cuadrático medio • Error cuadrático medio relativo • Raíz del error cuadrático medio relativo • Coeficiente de correlación
Agrupación de tipologías de clientes para una campaña de marketing	Agrupación o segmentación (*clustering*)	• Grado de cohesión dentro de los grupos en relación a la distancia entre grupos • Índice de Davies-Bouldin
Identificar libros que se compran conjuntamente en un comercio electrónico	Asociación	• Soporte y confianza • *Lift*

2.1. Clasificación

La métrica más común en modelos de clasificación es la precisión (*accuracy*). La precisión mide el porcentaje de aciertos, es decir, el número de clasificaciones realizadas correctamente en relación al total de clasificaciones efectuadas.

Tomemos un ejemplo sencillo del conjunto de datos iris disponible en el repositorio UCI (Lichman, 2013). En este problema, se deben clasificar instancias referidas a flores en tres tipos, *Iris setosa*, *Iris versicolor* o *Iris virginica*, a partir de los atributos de longitud y anchura del pétalo y sépalo. La precisión es el número de flores clasificadas correctamente en relación al total. Si se detalla la predicción por cada clase, se obtiene la matriz de confusión que se muestra en la tabla 2.

Tabla 2. Matriz de confusión para el conjunto de datos iris

		Clase real			
		Iris setosa	*Iris versicolor*	*Iris virginica*	Total
Predicción	*Iris setosa*	7	1	0	8
	Iris versicolor	1	8	0	9
	Iris virginica	2	1	10	13
	Total	10	10	10	30

La precisión se corresponde con el número de clasificaciones correctas (suma de valores de la diagonal) dividido por el total de instancias. Según los valores de la tabla, la precisión es: (7 + 8 + 10) / 30 que corresponde al porcentaje 83,3 %. También es interesante analizar la clasificación por cada clase o *true positive rate* (TPR). En la clase *Iris setosa*, TPR = 7 / 10, es decir, se predicen correctamente 7 de las 10 flores de tipo *Iris setosa* del conjunto de datos.

Es frecuente usar una métrica menos optimista de la precisión del modelo, denominada Kappa (Cohen, 1960). Esta métrica calcula el porcentaje de aciertos más allá del que se podría conseguir haciendo predicciones meramente al azar. Su formulación es:

$$Kappa = \frac{precisión - precisión\ \ al\ \ azar}{1 - precisión\ \ al\ \ azar}$$

Se puede medir la precisión del clasificador si hubiera realizado las clasificaciones al azar de la siguiente forma. Por ejemplo, para la flor *Iris setosa*, la probabilidad de que una flor sea de este tipo es 10 / 30 y la probabilidad de que el clasificador escoja iris-setosa es 8 / 30. Si el clasificador es al azar, la proba-

bilidad de acierto del mismo se obtiene a partir de la multiplicación de las dos probabilidades: $10 \cdot 8 / 30^2$. Por tanto, haciendo el cálculo para cada tipo de flor, la precisión de un clasificador al azar sería:

$$\frac{8 \cdot 10 + 9 \cdot 10 + 13 \cdot 10}{30^2} = \frac{300}{900} = 0{,}333$$

El valor Kappa es:

$$Kappa = \frac{0{,}833 - 0{,}333}{1 - 0{,}333} = 0{,}75$$

Como se observa, si restamos la contribución del azar en la clasificación, el clasificador pasa de un porcentaje de acierto aparente del 83,3 % a un porcentaje del 75 %.

A partir de la matriz de confusión se pueden extraer otras métricas como la sensitividad y especificidad, cuando el problema de clasificación es binario (dos clases). Hay otras consideraciones a tener en cuenta cuando se usan métricas de precisión. Por ejemplo, si el problema está muy desbalanceado la precisión no es una métrica adecuada de la calidad del modelo.

Predicción de fraude

Tomando como ejemplo el problema de predicción de fraude en tarjetas de crédito, este puede estar desbalanceado, ya que las transacciones no fraudulentas superan en número a las fraudulentas. Si el porcentaje de transacciones fraudulentas es del 99,5 %, un modelo que siempre prediga la clase mayoritaria tendría una precisión del 99,5 %.

Nota

Para más detalle sobre estos aspectos, el estudiante puede consultar el trabajo de Stapor (2017).

2.2. Regresión

En modelos de regresión se predice un valor numérico en lugar de una clase que pertenece a un conjunto finito de categorías. En este caso, no tiene sentido hablar de precisión, puesto que no se clasifica la instancia como perteneciente a una categoría u otra. Algunos ejemplos de regresión son la predicción del volumen de ventas de un producto o la predicción del área quemada de bosque (Lichman, 2013).

La calidad del modelo es el grado de aproximación del valor predicho al valor real. Se dispone de un conjunto de valores predichos p_1, p_2 ... p_n y sus correspondientes valores reales a_1, a_2 ... a_n. La métrica más básica es calcular la suma de las diferencias entre p_i y a_i en valor absoluto y dividirlas por n (el número de predicciones realizadas). Esta es la medida del error absoluto medio. Sin embargo, la medida más habitual es la raíz del error cuadrático medio (*Root Mean-Squared Error*), que se calcula según la fórmula siguiente:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (p_i - a_i)^2}{n}}$$

El inconveniente de la medida es que tiende a exagerar los efectos de los valores extremos. Además, un valor dado no tiene significado de forma aislada y debe contextualizarse en relación a los valores de predicción. No es lo mismo un error de valor 50 en relación a predicciones alrededor de 100 que sobre predicciones alrededor de 10.000. La raíz del error cuadrático medio relativo divide el valor en relación a los valores que se manejan. También pueden usarse otras métricas como el coeficiente de correlación de Pearson entre los valores predichos y los valores reales. Para más detalles sobre métricas de regresión, se puede consultar el libro de Witten, Frank y Hall (2011).

2.3. Agrupación

La evaluación de los métodos de agrupación o segmentación es más compleja que en clasificación o regresión porque no se dispone de un valor de referencia con el que comparar. Un criterio clásico es medir la calidad de la agrupación en base al grado de cohesión dentro del grupo en relación a las distancias entre grupos.

El índice de Davies-Bouldin (1979) presenta una medida de cohesión frente a distancia:

$$DB = \frac{1}{k}\sum_{\substack{i=1 \\ i\neq j}}^{k} \max\left(\frac{\sigma_i + \sigma_j}{d\ (C_i, C_j)}\right)$$

donde k es el número de grupos, C_x denota el centroide de cada grupo, σ_x es la distancia media de todos los elementos del clúster a su centroide respectivo C_x y $d\ (C_i, C_j)$ es la distancia entre los centroides C_i y C_j. Cuanto menor sea el valor de este índice, mejor agrupación.

2.4. Asociación

La identificación de patrones de asociación entre los atributos de un conjunto de datos es, al igual que en la agrupación, un caso en que no existe un valor de referencia. Las medidas de calidad se basan en criterios de frecuencia, representatividad y novedad de las asociaciones identificadas.

Por ejemplo, supongamos que se analizan las reglas de asociación en la compra de libros en una librería por internet. La sintaxis típica de las reglas de asociación consiste en un antecedente X y un consecuente Y. Así la regla siguiente:

Statistics, *Data analytics* → *Statistics for big data*

se lee como: «Si el cliente compra el libro *Statistics* y el libro *Data analytics*, entonces comprará también *Statistics for big data*».

Se usan dos métricas habituales para medir la calidad de la asociación: el soporte y la confianza. El soporte es el porcentaje de compras donde estos tres libros se compran conjuntamente:

$$\text{Soporte}\ \ (X \rightarrow Y) \ \ = \ \ \text{soporte}\ \ (\text{X} \cup \text{Y})$$

donde soporte(X∪Y) se refiere a la unión de los elementos que forman parte del antecedente y del consecuente. Por ejemplo, si entre 1.000 compras, 75 de ellas contienen estos tres libros a la vez, entonces el soporte es 75 / 1000 = 0,075 (7,5 %).

La confianza se calcula:

$$\text{Confianza}\ \ (X \rightarrow Y) \ \ = \ \ \frac{\text{soporte}\ \ (X \cup Y)}{\text{soporte}\ \ (X)}$$

En el ejemplo que nos ocupa, hemos visto que se compran 75 veces entre 1.000 los tres libros mencionados. Supongamos que en 120 ocasiones se compran los libros *Statistics* y *Data analytics*. Así pues, su confianza es 75 / 120 = 0,625.

La confianza puede interpretarse como la estimación de la probabilidad de encontrar el consecuente si se produce el antecedente. Es decir, la probabilidad condicionada p(Y|X).

Otra medida es el *lift* que compara la frecuencia de un patrón observado en relación a la expectativa de encontrar este patrón por azar. Se calcula como sigue:

$$\text{Lift} = \frac{\text{soporte}\ \ (X \cup Y)}{soporte\ \ (X)\ soporte\ \ (Y)}$$

Valores de *lift* cercanos a 1 implican una alta probabilidad de encontrar el patrón por azar en el conjunto de datos.

3. Evaluación del modelo

Cuando se evalúa un modelo de análisis de datos, el objetivo es tener una medida de la calidad del modelo cuando se use en explotación o producción. Por ejemplo, si un modelo predice el número de ventas de un determinado producto cada día, se espera que cuando el sistema esté implantado, la predicción que realice sea precisa. En general, el modelo no será perfecto, y por ello, es necesario conocer qué precisión (o margen de error) tendrá el sistema cuando se esté usando en su operación diaria. Además, hay un segundo motivo por el cual es necesario evaluar la calidad del modelo: cuando es necesario ajustarlo para máximo rendimiento, ya sea parametrizando adecuadamente el modelo elegido y/o escogiendo el mejor modelo entre varios disponibles.

En el ciclo de vida de la analítica de datos, existen dos fases bien diferenciadas:

- el **entrenamiento** del modelo
- el **testeo**

En la fase de entrenamiento se construye el modelo (corresponde a la fase 3 de la metodología presentada en el apartado 1). En la fase de testeo, se evalúa el modelo (fase 4). La construcción del modelo se realiza sobre una muestra de datos. Pero ¿cómo se debe realizar el testeo? Una opción sería realizar este testeo sobre la misma muestra de datos. Pero como adivinará el lector, este método va a resultar en una estimación sesgada positivamente, es decir, la evaluación será optimista. Y la causa de ello es que mientras evaluamos la calidad del modelo en el mismo conjunto de datos que se ha usado para construir el modelo, este modelo se usará en datos desconocidos en el momento de la producción. Consecuentemente, el modelo será menos preciso cuando opere con datos reales. Cabe decir que la diferenciación entrenamiento-testeo existe principalmente en modelos de tipo clasificación y regresión y son poco habituales en los de agrupación y asociación, ya que en los dos últimos no existe un valor de referencia con el que comparar.

En este apartado se define el problema de la estimación de la evaluación del modelo y se revisan los principales estimadores.

3.1. Estimación de la precisión de un modelo

El objetivo de testar un modelo es conocer su precisión o error cuando el modelo opere en datos reales. La precisión o error son *desconocidos a priori*, puesto que no se disponen de los datos con los que el sistema operará realmente. En adelante, se usará el término *precisión* refiriéndonos a la métrica de interés que se desee evaluar. Esta puede ser cualquiera de las métricas introducidas en el apartado 2. Puesto que la precisión real se desconoce *a priori*, tan solo se puede

realizar una estimación de la misma a partir del conjunto de datos disponibles durante la construcción del modelo. Llamamos a estos dos valores *precisión calculada en la muestra* y *precisión real*. Puesto que la estimación se realiza sobre una muestra de la población, existirá un *sesgo* y una *varianza*.

3.2. Sesgo y varianza del estimador

El estimador se puede considerar una variable aleatoria *Y* que estima un parámetro *p* de una población. El sesgo del estimador se define como la diferencia entre la esperanza de la variable Y y el parámetro a estimar:

$$E\ (Y) - p$$

La varianza de la variable aleatoria se define como:

$$Var(Y) = E((Y\text{-}p)^2)$$

La varianza se puede interpretar como la amplitud o dispersión de la distribución alrededor de la media.

3.3. Estimador basado en la muestra de entrenamiento

La fórmula más sencilla para estimar la precisión de un modelo es calcularla sobre la muestra de datos disponibles. Es decir, se estima la precisión real a partir de la precisión obtenida por el modelo sobre el conjunto de datos de entrenamiento, o sea, los mismos datos con los que se ha entrenado.

Como el conjunto de datos de la muestra suele ser pequeño, la generalización de la precisión calculada sobre la muestra a la precisión de la población estará sesgada positivamente. Es decir, la precisión en el conjunto de entrenamiento será mayor que la que el sistema tendrá en explotación (cuando se aplique a instancias o casos no usados directamente en el entrenamiento). Por tanto, decimos que la precisión de entrenamiento es un estimador sesgado optimistamente. Estimar la precisión en un conjunto de datos no usados en el entrenamiento es un mejor estimador de la precisión real.

3.4. Método de retención

El método de retención (*holdout*) divide el conjunto de datos disponibles en dos subconjuntos: el conjunto de entrenamiento y el conjunto de prueba (test). Se asume que los dos subconjuntos de datos son muestras representativas de la población de interés. El modelo se construye con el conjunto de datos de entrenamiento y luego se estima la calidad del mismo en el conjunto

de test. De esta forma, el modelo se testea en un conjunto de datos distinto al del entrenamiento. Este estimador no está sesgado optimistamente como sucede con el estimador basado en los datos de entrenamiento.

Habitualmente, el tamaño recomendado para estos subconjuntos es del 70 % de los datos en entrenamiento y el 30 % de datos para el test. El siguiente código R usa la función *createDataPartition* del paquete *caret* para crear dos subconjuntos de datos a partir del conjunto original:

```
intrain<-createDataPartition(y=data$class,  p=0.70)
train <- data [intrain,]
test <-  data [-intrain,]
```

El ejemplo está pensado para separar datos según *holdout* asumiendo que estos pertenecen a un problema de clasificación. La variable *data* es la muestra de datos y *class* es la clase asociada a los ejemplos de la muestra. En el subapartado 3.6 se verá por qué se introduce la variable *class* en la partición. La función *createDataPartition* devuelve los índices de los datos seleccionados, correspondientes a una muestra aleatoria del 70 % de los datos. Con ello, se construye la muestra *train* y con los datos restantes (los que no están indexados por *intrain*) se construye la muestra *test*.

El estimador *holdout* presenta varios inconvenientes. En primer lugar, si se dispone de una muestra de datos pequeña, reservar algunos datos para el testeo implica que los datos de entrenamiento se ven reducidos. Consecuentemente, la calidad del modelo construido será peor al no poder entrenar con todos los datos disponibles. El estimador estará sesgado negativamente. Otro inconveniente del método es que posee una elevada varianza. Si se repite el experimento con un muestreo distinto de datos en el entrenamiento y en el test, el error obtenido diferirá significativamente entre una prueba y otra.

Si el tamaño de los datos disponibles es suficientemente grande, el estimador presenta menor sesgo y varianza que si los datos son escasos y su uso podría ser aceptable. Para muestras pequeñas, una alternativa para minimizar el sesgo es realizar varias pruebas repetidas de *holdout*. El método de retención repetida (*repeated holdout*) realiza varios muestreos de conjuntos entrenamiento-test y para cada uno entrena el modelo y evalúa su precisión. La precisión final estimada será el promedio de la precisión cometida en cada muestreo.

El método de retención repetida nos introduce en el mundo del remuestreo (*resampling*) que pasamos a explicar a continuación.

3.5. Remuestreo

Las técnicas de remuestreo (*resampling*) consisten en muestrear múltiples veces el conjunto de datos original para obtener distintos pares de subconjuntos de entrenamiento y test con el objetivo de obtener mejores estimadores.

3.5.1. Validación cruzada

La validación cruzada (*cross-validation*) consiste en dividir el conjunto de datos en k particiones disjuntas denominadas *folds*. El método se entrena k veces. Para la iteración i donde $1 \leq i \leq k$, se construye el modelo a partir del subconjunto de datos formado por todas las particiones excepto la i, y se testea el modelo en el subconjunto i. La precisión final es el promedio de la precisión obtenida en los k testeos. Con ello se consigue que el método se testee con todos los ejemplos disponibles y a la vez se estrene con instancias independientes del testeo. Un valor habitual de k es 10 (*10-fold cross-validation*), con lo que se entrena en cada ocasión con el 90 % de la muestra original. La tabla 3 especifica un ejemplo de ejecución de este método de remuestreo para el problema de clasificación de flores iris (Lichman, 2013).

Tabla 3. Estimación de la precisión mediante el método de validación cruzada (k=10)

	Subconjunto de entrenamiento	**Subconjunto de test**	**Precisión (acc)**
1	{2-10}	1	0,92
2	{1, 3-10}	2	0,99
3	{1-2, 4-10}	3	0,98
4	{1-3, 5-10}	4	0,89
5	{1-4, 6-10}	5	0,94
6	{1-5, 7-10}	6	0,96
7	{1-6, 8-10}	7	0,98
8	{1-7, 9-10}	8	0,95
9	{1-8, 10}	9	0,93
10	{1-9}	10	0,97
Precisión media ($\overline{acc}$)			0,951
Desviación estándar (SD)			0,031

El siguiente código ilustra cómo ejecutar el método de vecinos más próximos k-NN con validación cruzada 10-CV:

```
ctrl<-trainControl(method="repeatedcv", repeats=1)
knn<-train(Species~., data=iris, method="knn",
preProc=c("center","scale"),
trControl=ctrl)
knn
## k-Nearest Neighbors
##
## 150 samples
##   4 predictor
##   3 classes: 'setosa', 'versicolor', 'virginica'
##
## Pre-processing: centered (4), scaled (4)
## Resampling: Cross-Validated (10 fold, repeated 1 times)
## Summary of sample sizes: 135, 135, 135, 135, 135, 135, ...
## Resampling results across tuning parameters:
##
##   k  Accuracy   Kappa
##   5  0.9600000  0.94
##   7  0.9533333  0.93
##   9  0.9533333  0.93
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 5.
```

Como se puede observar, se usan 10 particiones y cada una de ellas tiene 135 datos (de los 150 de la muestra original) en el conjunto de entrenamiento. El código prueba tres valores del parámetro *k* (número de vecinos) del método k-NN, ejecutando una validación cruzada para cada valor del parámetro.

Nota

k-NN es el nombre que recibe el método de minería de datos denominado vecinos más cercanos. En este, el parámetro k configura el número de vecinos que extrae el método para devolver su resultado. No confundir con el parámetro k del método de validación cruzada, que se refiere al número de particiones de la muestra de datos. Ambos métodos usan el mismo nombre para sus respectivos parámetros.

Para un número de particiones *k* elevado, el sesgo del método es inferior al método de retención, puesto que el modelo se entrena con un porcentaje elevado de los datos (para *k* = 10, se entrena con un 90 % de los datos). Sin embargo, sigue siendo un estimador sesgado negativamente, puesto que el modelo se entrena con un (*k* - 1) / *k* 100 % de los datos, y se espera que será peor que un modelo entrenado con todo el conjunto de datos. El sesgo puede minimizarse con el método LOO que se explica a continuación.

Además del sesgo, los resultados de la validación cruzada tienen una elevada varianza. Si se ejecutan dos pruebas de validación cruzada con particiones distintas, los resultados de cada una de ellas serán distintos. Para minimizar la varianza, se puede repetir la validación cruzada varias veces con distintos submuestreos. En este caso, el método se llama validación cruzada repetida (*repeated cross-validation*). El inconveniente es el elevado coste computacional que supone entrenar el modelo repetidas veces (concretamente $n \cdot k$ veces, donde n son las veces que se repite y k es el número de particiones).

3.5.2. LOO

Un caso particular del método de validación cruzada es el denominado *Leave-One-Out* (LOO). En este método cada muestra de test se corresponde con un único ejemplo. Así, el modelo se construye n veces, donde n es el número de instancias de la muestra original. En cada iteración, el modelo se entrena con n-1 ejemplos y se testea con el ejemplo restante. La ventaja del LOO es que el modelo se construye con prácticamente todos los ejemplos disponibles (n-1), a la vez que se testea con todos los ejemplos de test, al igual que el testeo en la validación cruzada. No obstante, el elevado coste computacional hace que

sea un método poco aplicable excepto en aquellos casos que se disponga de un conjunto de datos muy reducido. El código siguiente muestra la aplicación del método LOO usando R:

```
ctrl<-trainControl(method="LOOCV", repeats=1)
knn<-train(Species~., data=iris, method="knn",
preProc=c("center","scale"),
trControl=ctrl)
knn
## k-Nearest Neighbors
##
## 150 samples
##   4 predictor
##   3 classes: 'setosa', 'versicolor', 'virginica'
##
## Pre-processing: centered (4), scaled (4)
## Resampling: Leave-One-Out Cross-Validation
## Summary of sample sizes: 149, 149, 149, 149, 149, 149, ...
## Resampling results across tuning parameters:
##
##   k  Accuracy   Kappa
##   5  0.9466667  0.92
##   7  0.9600000  0.94
##   9  0.9533333  0.93
##
## Accuracy was used to select the optimal model using  the largest value.
## The final value used for the model was k = 7.
```

3.5.3. *Bootstrap*

El método *bootstrap* se basa en el remuestreo con reemplazo. Los anteriores remuestreos se realizaban sin reemplazo: una vez un ejemplo era seleccionado para un subconjunto, este no podía ser seleccionado de nuevo. Ello generaba conjuntos disjuntos de entrenamiento y test. En *bootstrap*, cada dato puede ser seleccionado varias veces para formar parte del subconjunto de entrenamiento. En el denominado método *0,632 bootstrap*, se seleccionan n instancias de un conjunto de datos de tamaño n para formar parte de un conjunto de entrenamiento, que será del mismo tamaño que el original. Como algunas instancias no saldrán seleccionadas, se usan estas para formar parte del conjunto de prueba. El nombre del método proviene de la probabilidad de que una instancia no forme parte del conjunto de entrenamiento, que se calcula como:

$$\left(1-\frac{1}{n}\right)^n \approx e^{-1} = 0{,}368$$

Así pues, el tamaño del conjunto de test es del 36,8 % y el conjunto de entrenamiento el 62,3 % de la muestra original aproximadamente.

El estimador es pesimista porque el conjunto de entrenamiento solo contiene el 62,3 % de las instancias. Para compensarlo, la precisión final se calcula como:

$$\text{acc} = 0{,}632 \cdot \text{acc}_{test} + 0{,}368 \cdot \text{acc}_{train}$$

El siguiente código en R ejecuta el método k-NN usando *bootstrap*:

```
ctrl<-trainControl(method="boot632")
knn<-train(Species~., data=iris, method="knn",
preProc=c("center","scale"))
knn
## k-Nearest Neighbors
##
## 150 samples
##   4 predictor
##   3 classes: 'setosa', 'versicolor', 'virginica'
##
## Pre-processing: centered (4), scaled (4)
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 150, 150, 150, 150, 150, 150, ...
## Resampling results across tuning parameters:
##
##   k  Accuracy   Kappa
##   5  0.9427889  0.9136946
##   7  0.9359550  0.9033683
##   9  0.9410505  0.9110650
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 5.
```

3.6. Estratificación

Al introducir el método de retención (*holdout*) se ha especificado que las muestras de entrenamiento y prueba deben ser representativas de la población de interés. En primer lugar, la muestra original de datos debería ser representativa. Si no lo es, al hacer el remuestreo en los dos subconjuntos de entrenamiento y testeo se preservará la misma falta de representatividad. En el módulo 1, se ha mencionado el muestreo probabilístico para seleccionar una muestra de datos representativa. Pero aun disponiendo de una muestra original representativa, puede suceder que al remuestrear en los dos subconjuntos se produzcan sesgos. Tomando como ejemplo el problema de clasificación iris, podría suceder que el conjunto de entrenamiento solo contenga muestras de dos de las tres flores, lo cual imposibilitaría extraer un modelo que clasifique la flor que no está representada. Además, la flor que no está presente en el conjunto de entrenamiento, estaría sobrerrepresentada en el conjunto de testeo, con lo cual el error de clasificación sería elevado.

El problema se presenta especialmente con muestras pequeñas. Aunque el método de remuestreo garantice que cada instancia del conjunto de datos tiene igual probabilidad de ser seleccionada en el subconjunto de entrenamiento, pueden existir sesgos inevitables. La solución pasa, al igual que en el muestreo de la muestra original, por realizar un muestreo estratificado. En este, los estratos son cada una de las clases. En el caso del problema iris, se separan las instancias en tres grupos, uno para cada clase. Luego, se realiza un muestreo aleatorio dentro de cada clase. Así, se garantiza que en los subconjuntos de entrenamiento y de test se preserva la proporción de clases del conjunto de datos original. Y con esto damos también respuesta a la línea de código especificada anteriormente, cuando se introdujo el método *holdout*:

```
createDataPartition( y=data$class, p=0,70 )
```

Como se puede deducir, se indica al método de partición *holdout* que considere la clase asociada a los ejemplos del conjunto *data* para aplicar estratificación. La estratificación puede aplicarse en todos los métodos de remuestreo.

4. Validación estadística

En la sección anterior, se ha tratado la estimación de la calidad o rendimiento de los métodos de análisis de datos medida sobre una muestra de datos. Dado que habitualmente el número de datos para entrenar el algoritmo y testear su calidad es limitado, se han introducido los métodos de remuestreo de datos con el objetivo de obtener una estimación precisa de la calidad del modelo. Una vez se dispone de esta estimación, se plantea la cuestión de cómo proceder a partir de este momento y ello dependerá del contexto y objetivo del diseño experimental. Los escenarios posibles son:

- Generalización de los resultados
- Comparación entre varios modelos de datos en un problema dado
- Comparación entre varios modelos de datos en múltiples problemas

1) Generalización de los resultados. El primer punto se refiere a estudiar cómo se generalizan los resultados a nuevos datos. Es decir, consiste en inferir la estimación de la calidad del modelo obtenida sobre la muestra a la calidad que el modelo tendría en explotación, cuando se usa en datos futuros, que pertenecen a la población de interés. La formulación general sería:

> Dado un conjunto de valores de precisión acc_1, acc_2 ... acc_L, cada uno de ellos referido a un subconjunto de test, ¿cuál sería la precisión *acc* del modelo cuando se aplica a datos reales?

En términos estadísticos, consiste en inferir el valor del parámetro poblacional a partir del parámetro muestral. Así, se considera cada valor acc_i como el valor de un individuo de una muestra. Se calcula la media y desviación muestral y a continuación, se deduce el **intervalo de confianza** de la media de la población. El método no está exento de debate, puesto que si los valores acc_i provienen de muestras no independientes, como el caso de la validación cruzada repetida, no sería apropiado. Se recomienda entonces calcular los intervalos de confianza a partir de *bootstrap*. La profundización en este debate está fuera del alcance de este módulo. El estudiante puede consultar el trabajo de Vanwinckelen y Blockeel (2012) para profundizar en este aspecto.

2) Comparación entre modelos en un problema dado. El segundo escenario es muy habitual en experimentación científica de datos, así como en ámbitos aplicados. Se trata de comparar el rendimiento de dos o más métodos en un problema dado. La formulación sería:

> Dados *K* modelos distintos, ¿cuál de ellos ofrece un mejor rendimiento para el problema dado?

Estos modelos distintos pueden representar distintos algoritmos de análisis de datos, como el k-NN y los árboles de decisión, o pueden provenir de un mismo algoritmo con distintos parámetros de configuración (por ejemplo, valores distintos del número de vecinos *k* para el algoritmo k-NN). En términos estadísticos, la pregunta formulada se traduce en un **contraste de hipótesis para dos o más muestras**. Si los distintos modelos se comparan bajo las mismas condiciones (el mismo remuestreo de los datos, las mismas particiones y las mismas métricas), entonces podemos considerar que las muestras están apareadas. En este caso, se aplican **contrastes de hipótesis para dos o más muestras apareadas**. Es necesario hacer la distinción según si comparan dos algoritmos o más de dos. La elección de los métodos de validación estadística también diferirá según estos dos casos. Una última consideración es decidir entre tests paramétricos y tests no parámetricos. La tabla 4 resume los principales tests estadísticos.

Tabla 4. Pruebas estadísticas para la comparación de modelos de análisis de datos

Muestras	Comparación	Tests paramétricos	Tests no paramétricos
Apareadas	Dos modelos	• Test t para muestras apareadas • Test z	• Prueba de la suma de rangos de Wilcoxon • Test de signo • Test de McNemar
	Más de dos modelos	• Análisis de varianza (ANOVA) • Test z	• Prueba de Kruskall-Wallis • Prueba de Friedman

3) Comparación entre modelos en múltiples problemas. Este escenario es la generalización del segundo escenario a múltiples problemas. Pertenece básicamente al dominio de los estudios científicos más que a dominios aplicados. El interés inherente es encontrar modelos de análisis de datos que ofrezcan rendimientos superiores a los existentes. Para ello, el experimentador testea el modelo en relación a otros modelos estándar de análisis de datos y extrae conclusiones generales sobre el rendimiento del modelo en comparación con los otros. Otro objetivo podría ser el de comparar el rendimiento de modelos existentes, sin un interés destacado en ninguno de ellos *a priori* (Lim, Loh y Shih, 2000). En cambio, en el ámbito aplicado, el interés más común es desarrollar el modelo óptimo para un problema dado.

La formulación general de la comparación de múltiples modelos en múltiples problemas es:

Dados *N* problemas y *K* modelos, ¿cuál de los modelos ofrece un mejor rendimiento?

La formulación requiere algunos matices. En primer lugar, puede que no exista un método que tenga un rendimiento general mejor. Por este motivo, primero se formula si existen diferencias entre todos los métodos. *A posteriori*, si hay evidencia de diferencias entre ellos, se identifica cuál de los métodos es el que destaca sobre los demás. Este tipo de formulación parte de una hipótesis

de que un determinado método es mejor que los demás. Y ello surge de un estudio experimental de ciencia de datos, en que el investigador ha propuesto un nuevo método que se contrasta con otros existentes.

Al trasladar esta formulación en términos de validación estadística, encontramos el mismo tipo de aproximaciones que para la comparación de métodos en un problema dado. La distinción está en si se comparan dos o más métodos, y si se pueden aplicar tests paramétricos o no paramétricos, tal como se muestra en la tabla 4. Las diferencias metodológicas entre los dos escenarios se refieren a la obtención de los valores de calidad de cada modelo. Para un problema dado (escenario 2), los valores a comparar provienen de los distintos submuestreos de datos (*cross-validation*, *bootstrap*...), tal como se ha detallado en el apartado anterior. Para múltiples problemas, los valores que se comparan provienen de la precisión media del modelo en diferentes problemas. Esta diferencia puede determinar la elección de los tests estadísticos apropiados (Demsar, 2006).

4.1. Comparación entre dos modelos

En esta sección se describen los principales métodos de validación estadística aplicables a la comparación de dos modelos de datos. Se asume que los modelos arrojan respectivamente unas medidas de precisión x_1, x_2, ... x_N e y_1, y_2, ... y_N. Estos datos conforman las dos muestras a comparar, asumiendo que son dos muestras apareadas. Los métodos que veremos a continuación son aplicables a la comparación entre dos modelos para un problema dado o para múltiples problemas. Como se ha comentado, la diferencia está en que los datos de precisión provienen de los distintos submuestreos de datos (para un problema dado) o de distintos problemas. Al final de esta sección, se detallan los matices existentes según si la comparación entre los dos modelos se produce sobre un único problema o sobre varios problemas.

4.1.1. Test *t* para dos muestras apareadas

El lector probablemente ya conoce el test t apareado. Es un test muy habitual en diseños experimentales. Aplicado al diseño experimental de modelos de datos, asumimos que se dispone de los resultados de dos modelos en un conjunto de *N* problemas: x_1, x_2, ... x_N e y_1, y_2, ...y_N. El test *t* verifica si la diferencia en el rendimiento sobre los conjuntos de test es significativamente distinta de cero.

Se puede asumir que las muestras están apareadas si los resultados de cada modelo se han computado sobre los mismos conjuntos de datos. Así, se calculan las diferencias entre los dos modelos: $d_i = x_i - y_i$. La media de las diferencias

es la diferencia de las medias: $\overline{d} = \overline{x} - \overline{y}$ y se comporta como una distribución *t* con *N*-1 grados de libertad. La hipótesis nula establece que la media de las diferencias es cero. El estadístico *t* se calcula según:

$$t = \frac{\overline{d}}{\sqrt{S_d^2 / N}}$$

donde $\overline{d}$ es la media de las diferencias, S_d^2 es la varianza de las diferencias y *N* el tamaño de la muestra.

El test t presenta dos restricciones. La primera es que asume una distribución normal de los datos, lo cual habitualmente no se cumple. Por otra parte, el test requiere que los datos sean independientes. Si se aplica para la comparación de dos métodos en un único problema, esta asunción no se cumple (Salzberg, 1997). El motivo es que los tests de entrenamiento y de prueba no son totalmente independientes, como sucede por ejemplo si se usa una validación cruzada repetida. La alternativa es usar tests no paramétricos como los que se describen a continuación.

4.1.2. Prueba de suma de rangos de Wilcoxon

Demsar (2006) recomienda usar la prueba de suma de rangos de Wilcoxon (1945) o el test de signo. Estos son los equivalentes no paramétricos del test t apareado, tal como se muestra en la tabla 4. A continuación, se explica la aplicación de la prueba de Wilcoxon puesto que su uso está más extendido.

Se aplica tal como sigue:

1) Se calculan las diferencias entre los dos métodos a comparar.

2) Estas diferencias se ordenan en orden creciente y en valor absoluto. Se asigna un rango a cada valor, en función de su posición. En caso de empate, se calcula el rango promedio.

3) Se calcula *R*+ como la suma de los rangos en que el primer modelo mejora el segundo. De forma inversa, se calcula R-. Si una distancia es igual a 0, se reparte entre *R*+ y *R*-.

4) Se calcula *T* = mínimo (*R*+ ,*R*-). Su valor crítico se puede calcular a través de tablas (hasta *N* igual a 25). Para valores mayores de *N*, se usa el estadístico *z* que se distribuye normalmente:

$$z = \frac{T - \frac{1}{4}N(N+1)}{\sqrt{\frac{1}{24}N(N+1)(2N+1)}}$$

En la tabla 5 se detalla un ejemplo para dos conjuntos de muestras.

Tabla 5. Ejemplo de cálculo del test de suma de rangos de Wilcoxon

x	y	d	rango
10,54	12,04	-1,50	-10
10,70	11,75	-1,05	-8
10,23	11,22	-0,99	-7
10,43	10,18	0,25	3
10,53	11,34	-0,81	-6
10,98	9,73	1,25	9
10,62	10,67	-0,05	-1
10,81	11,11	-0,30	-4
10,40	10,24	0,16	2
10,50	10,87	0,37	-5

La suma de rangos positivos $R+$ es igual a 14 y la de rangos negativos $R-$ es igual a 41. Por tanto, $T = min\ (R+ ,R-) = 14$. El valor de z es -1,38 y el valor p correspondiente de las tablas de distribución normal en una prueba bilateral es 0,1688, con lo que no se puede rechazar la hipótesis nula de que no existen diferencias entre los dos modelos.

El código siguiente muestra el cálculo de la prueba de Wilcoxon en *R*:

```
wilcoxsign_test(x1~x2)
##
##  Asymptotic Wilcoxon-Pratt Signed-Rank Test
##
## data:  y by x (pos, neg)
##   stratified by block
## Z = -1.376, p-value = 0.1688
## alternative hypothesis: true mu is not equal to 0
```

Para valores pequeños de N (tamaño de la muestra) es más recomendable consultar tablas de cálculo directo del valor p en lugar del valor z, puesto que ofrece mejor aproximación.

4.1.3. Prueba de McNemar

La prueba de McNemar es de tipo no paramétrico y se aplica comparando el resultado de los dos modelos para cada instancia del conjunto de datos. Se construye una tabla como la siguiente:

Tabla 6. Cálculo del test de McNemar

n00 Número de instancias mal clasificadas por los dos modelos	n01 Número de instancias incorrectamente clasificadas por el método X pero no por Y
n10 Número de instancias incorrectamente clasificadas por el método Ypero no por X	n11 Número de instancias clasificadas correctamente por los dos modelos

donde $n = n_{00} + n_{01} + n_{10} + n_{11}$ es el número total de ejemplos del conjunto de datos. El test está pensado para muestreos del tipo *holdout*. Bajo la hipótesis nula, los dos modelos deberían tener el mismo error ($n_{01} = n_{10}$). Para ello, se compara el número de errores bajo la hipótesis nula en relación a los valores observados:

$$\frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}}$$

Este valor se distribuye según χ^2 con 1 grado de libertad. Si la hipótesis nula es cierta, entonces la probabilidad de que este valor sea mayor que $\chi^2_{1,0.95} = 3{,}84$ es menor que 0,05. Por tanto, se rechazaría la hipótesis nula a favor de la hipótesis de que los dos modelos tienen rendimiento distinto.

Según Dietterich (1998), este test presenta bajo error tipo I. El error tipo I es la capacidad de detectar diferencias cuando estas no existen. Como inconveniente, el test aplicado con un muestreo *holdout* no tiene en cuenta la variabilidad debida a la elección del conjunto de entrenamiento ni la variabilidad debida al propio algoritmo. Dietterich recomienda usar este test solo cuando estas variabilidades son pequeñas.

4.1.4. Comparación de dos modelos en uno o varios problemas

Demsar (2006) argumenta que la comparación de dos modelos en un único problema es insegura debido a la falta de independencia entre los valores de precisión obtenidos, especialmente en aquellos casos en que los conjuntos de entrenamiento y prueba provienen de remuestreos iterativos. En estos casos, Demsar recomienda el uso de la prueba de McNemar, puesto que no realiza asunciones de distribución de los datos. Sin embargo, como se ha comentado, no tiene en cuenta la variabilidad en la elección del conjunto de entrenamiento ni la variabilidad interna del algoritmo de aprendizaje que extrae el modelo de los datos.

Si se comparan dos modelos en múltiples conjuntos de datos, las medidas son difícilmente comparables, puesto que los datos provienen de fuentes diversas. Para ello, Demsar recomienda el test de suma de rangos de Wilcoxon.

4.2. Comparación de múltiples modelos

En la comparación múltiple de modelos, se quieren comparar múltiples métodos *K* en *N* conjuntos de datos distintos. Para cada método y cada problema se dispone de una estimación de la precisión o calidad del modelo. Se descarta de momento el análisis de varianza sobre el remuestreo y se asume que el valor de rendimiento de cada método es una estimación precisa. Esta estimación puede provenir de un método de validación cruzada u otro método de remuestreo.

Al ejecutar algoritmos en múltiples problemas se obtienen medidas independientes, a diferencia de la comparación sobre un único problema. Por tanto, las comparaciones son más simples que en la comparación con el mismo conjunto de datos (Demsar, 2006).

4.2.1. Corrección de Bonferroni

Un error habitual al comparar múltiples modelos en varios conjuntos de datos es usar el test t de muestras apareadas (Kuzon, Urbancheck y McCabe, 1996). El problema que se deriva de usar el test t apareado en comparación múltiple es que el error de tipo I que se comete se propaga para cada comparación por pares y por tanto, se debería ajustar el nivel de significación α a un valor más restrictivo (Salzberg, 1997).

Este es el método que sigue la corrección de Bonferroni. Si se disponen de *m* hipótesis a contrastar, donde *m* es el número de pares de algoritmos que se comparan entre sí $m = K\ (K - 1)\ /\ 2$, el valor de α corregido sería el valor original de α dividido por *m*. Por ejemplo, para cuatro modelos a comparar con un valor de significación α igual a 0,05, el valor corregido de α es 0,05 / 6 que corresponde a 0,0083. Una vez corregido, se aplica el test t a cada par de métodos.

La corrección de Bonferroni es muy restrictiva, lo cual deriva en una prueba con poca potencia. Es decir, la capacidad para detectar diferencias significativas cuando estas existen es muy baja.

4.2.2. ANOVA

El método estadístico habitual para comparaciones múltiples es ANOVA o análisis de varianza. Las muestras apareadas corresponden a la calidad de cada modelo sobre los mismos conjuntos de datos, preferentemente usando las mismas particiones de los subconjuntos de entrenamiento y test. La hipótesis nula es que no existen diferencias entre los distintos métodos que se comparan.

ANOVA divide la variabilidad total en la variabilidad entre los modelos de datos, la variabilidad entre los conjuntos de datos y la variabilidad residual. Si la variabilidad entre modelos es suficientemente mayor, rechazamos la hipóte-

sis nula a favor de la hipótesis alternativa de que existen diferencias entre los modelos. Remitimos al estudiante a textos de estadística básica para conocer los detalles del test.

Si se rechaza la hipótesis nula, se puede decir que existen diferencias significativas entre los modelos, pero no se conoce cuáles de ellos son diferentes a otros. Para ello, se aplican pruebas denominadas *post-hoc* (*a posteriori*), también denominadas simplemente pruebas de comparaciones múltiples. Entre estos métodos está el test de Tukey para la comparación entre todos los modelos, o el test de Dunnett para comparar los modelos en relación a uno de ellos. Las dos pruebas son similares al test t, con la corrección de que los valores críticos son más elevados para asegurar que se mantiene el valor de α determinado (como sucede con la corrección de Bonferroni).

El inconveniente de ANOVA es que se asume que las muestras se distribuyen normalmente. Otro requisito es la esfericidad (homogeneidad de varianza). Si estos requisitos no se cumplen, el test de Friedman, que es el equivalente no paramétrico, es más aconsejable. A continuación, se describe con más detalle el test de Friedman puesto que su aplicación en el contexto del diseño experimental en analítica de datos está más extendida.

4.2.3. Test de Friedman

El test de Friedman ordena cada modelo en función de su rendimiento en cada problema y calcula el promedio de los rangos de cada modelo. El estadístico se calcula según la fórmula siguiente:

$$\chi_F^2 = \frac{12N}{K(K+1)}\left[\sum_j R_j^2 - \frac{K(K+1)^2}{4}\right]$$

donde R_j^2 es el promedio de los rangos de cada algoritmo, N es el número de problemas y K el número de modelos que se comparan. Para el cálculo de R_j, se ordena el resultado de cada modelo para cada problema, de forma que el modelo con mejor resultado obtiene el rango 1, el siguiente rango 2, etc. El rango total R_j de cada modelo es el promedio de los rangos del modelo en todos los problemas. Bajo la hipótesis nula, todos los modelos son equivalentes, es decir, todos los rangos R_j son iguales. El estadístico se distribuye según χ_F^2 con (K-1) grados de libertad.

Ivan-Davenport proponen una corrección sobre el estadístico para mejorar la potencia del test:

$$F_F = \frac{(N-1)\chi_F^2}{N(K-1)-\chi_F^2}$$

que se distribuye según F con K-1 y (K-1)(N-1) grados de libertad.

El test de Friedman tiene menos potencia que el test ANOVA cuando las condiciones de ANOVA se satisfacen. Cuando se rechaza la hipótesis nula de que todos los algoritmos son equivalentes, se pasa a realizar un test *post-hoc*.

4.2.4. Pruebas de comparaciones múltiples o pruebas a posteriori

Las pruebas a posteriori (*post-hoc*) realizan comparaciones por pares entre los modelos para identificar cuáles de ellos tienen mejor rendimiento. Las comparaciones por pares se asocian dentro de una familia de hipótesis y por tanto, es necesario controlar el valor de significación α. El estadístico para comparar dos modelos es:

$$z = \frac{(R_i - R_j)}{\sqrt{\frac{K(K+1)}{6N}}}$$

donde R_i y R_j son los rangos respectivos de cada método, K es el número de modelos que se comparan y N el número de problemas o conjuntos de datos.

Test de Nemenyi

El test de Nemenyi se usa para comparar todos los modelos entre sí, de forma análoga al test Tukey para ANOVA. Para ello, ajusta el valor de α dividiendo por el número de comparaciones que se realizan $m = K\ (K - 1)\ /\ 2$. Este procedimiento es el más simple, pero también el que tiene menos potencia.

Una formulación alternativa del test propuesta por Demsar (2006) consiste en calcular la distancia crítica DC:

$$DC = q_\alpha \sqrt{\frac{K(K+1)}{6N}}$$

Para calcular q_α se usa el estadístico de rangos Student para infinitos grados de libertad dividido por $\sqrt{2}$. La tabla 7 presenta los valores críticos para α = 0,05, en función del número de modelos que se comparan entre sí.

Tabla 7. Distancias críticas para el test de Nemenyi y test de Bonferroni-Dunn bilaterales

Número de modelos	2	3	4	5	6	7	8	9	10
Nemenyi $q_{0,05}$	1,960	2,343	2,569	2,728	2,850	2,949	3,031	3,102	3,164
Bonferroni − Dunn $q_{0,05}$ (α/(K-1))	1,960	2,241	2,394	2,498	2,576	2,638	2,690	2,724	2,773

Ajuste de α para la comparación con un modelo de referencia

Cuando todos los modelos se comparan con un modelo de referencia, se pueden usar otros procedimientos menos conservadores que se ajustan con *K*-1 al comparar con un único algoritmo de control en lugar del ajuste *K* (*K*-1) / 2. El procedimiento consiste en calcular el valor *z* a partir de los rangos entre los modelos y el modelo de referencia, tal como se ha descrito anteriormente. A continuación, se usa la tabla de distribución normal para obtener el valor de probabilidad *p*. Este valor se compara con el valor de α deseado.

Existen varias pruebas que ajustan el valor de α de formas distintas. A continuación, se describen las más habituales.

Prueba de Bonferroni-Dunn

Este test ajusta el error dividiendo α con el número de comparaciones (*K*-1) o alternativamente en el cálculo de distancia crítica (DC) con valores críticos α/(*K*-1). La corrección Bonferroni-Dunn es menos conservadora que el test de Nemenyi.

Prueba de Holm

En este test se ordenan las hipótesis por orden del valor de *p* correspondiente: $p_1<p_2< \ldots <p_{K\text{-}1}$. Entonces se compara cada p_i de la forma siguiente. Si p_1 es superior a α/(*K*-1), se rechaza la hipótesis correspondiente, y se pasa a comparar p_2 con α/(*K*-2). Así sucesivamente hasta que un valor p_i no permite rechazar más hipótesis, momento en que no se realizan más comparaciones.

Prueba de Hochberg

El test de Hochberg ordena los valores $p_1 < p_2 < \ldots < p_{K\text{-}1}$ y empieza con el valor mayor de *p* (p_{K-1}). Se compara el valor p_{K-1} con α, el siguiente valor p_2 con α/2 y así sucesivamente hasta que encuentra una hipótesis que pueda rechazar. Entonces, el resto de hipótesis con valores de *p* más pequeños son rechazadas también.

4.2.5. Ejemplo de comparación múltiple

Para ilustrar la aplicación de las pruebas de comparación múltiple, se realizará una comparación de $K = 4$ métodos con $N = 15$ conjuntos de datos. Se asume que no se cumplen las condiciones de aplicación de las pruebas paramétricas, con lo cual se aplicarán las pruebas no paramétricas equivalentes.

En la tabla 8 se presentan los resultados de la precisión de los cuatro modelos estudiados en los 15 problemas:

Tabla 8. Ejemplo de comparación múltiple. Resultados de 4 modelos en 15 problemas

Problema	M1	M2	M3	M4
1	94,97	90,82	97,91	97,22
2	97,66	113,04	94,01	88,33
3	98,41	107,01	97,73	92,19
4	91,50	103,82	96,17	84,75
5	88,69	83,59	92,39	94,21
6	91,45	97,94	96,52	93,99
7	92,61	91,10	96,19	93,72
8	93,70	93,85	96,82	94,35
9	91,36	99,17	100,09	93,26
10	88,65	90,67	94,48	89,50
11	93,46	91,88	94,22	89,05
12	90,51	98,68	94,91	92,50
13	90,50	99,2	99,67	90,19
14	92,77	95,51	96,25	99,16
15	92,95	96,97	97,84	78,98

En primer lugar, se calculan los rangos de cada modelo para cada problema. En la siguiente tabla se resumen estos rangos:

Tabla 9. Rangos de la comparación de los 4 modelos en los 15 problemas

Problema	M1	M2	M3	M4
1	3	4	1	2
2	2	1	3	4
3	2	1	3	4
4	3	1	2	4
5	3	4	2	1
6	4	1	2	3
7	3	4	1	2
8	4	3	1	2
9	4	2	1	3
10	4	2	1	3
11	2	3	1	4
12	4	1	2	3

Problema	M1	M2	M3	M4
13	3	2	1	4
14	4	3	2	1
15	3	2	1	4
Rango medio	**3,20**	**2,27**	**1,60**	**2,93**

Se calcula el test de Friedman:

$$\chi_F^2 = \frac{12 \cdot 15}{4 \cdot 5}\left[3{,}20^2 + 2{,}27^2 + 1{,}60^2 + 2{,}93^2 + \frac{4 \cdot 5^2}{4}\right] = 13{,}88$$

$$F_F = \frac{(15-1) \cdot 13{,}88}{15 \cdot (4-1) - 13{,}88} = 6{,}24$$

F_F se distribuye con (4-1) y (4-1)*(15-1) = 42 grados de libertad. El valor crítico de $F_{3,42}$ para α = 0,05 se puede consultar en las tablas de distribución *F* o alternativamente mediante *R*:

```
qf(0.05, k-1, (k-1)*(N-1), lower.tail=FALSE)
## [1] 2.827049
```

El valor crítico es 2,83. Como el valor obtenido *F* = 6,24 es mayor que el valor crítico, se concluye que existen diferencias significativas. Alternativamente, se puede consultar el valor *p* asociado al estadístico de contraste obtenido:

```
pf(F, k-1, (k-1)*(N-1), lower.tail=FALSE)
## [1] 0.001326882
```

Puesto que el valor *p* es inferior a 0,05, se descarta la hipótesis nula de que todas las diferencias son iguales. A continuación, se calcula la prueba de Nemenyi para comparar los modelos entre sí. La forma más sencilla es calcular la distancia crítica DC consultando el valor $q_{0,05}$ de la tabla 7:

$$DC = 2{,}569\sqrt{\frac{4 \cdot 5}{6 \cdot 15}} = 1{,}21$$

Para que existan diferencias entre dos pares de métodos, sus rangos deben diferir al menos en 1,21. En la tabla siguiente se muestran las diferencias de rangos entre cada par de métodos.

Tabla 10. Diferencias entre rangos del ejemplo de comparación múltiple

	1	2	3	4
1	0	0,93	**1,6**	0,27
2	-0,93	0	0,67	-0,67

3	**-1,6**	-0,67	0	**-1,33**
4	-0,27	0,67	**1,33**	0

Como se observa en la tabla, las diferencias entre rangos que superan el valor de distancia crítica 1,21 son las existentes entre el modelo 1 y 3 y entre el modelo 3 y 4. El resto de comparaciones entre pares de modelos da como resultado el no rechazo de la hipótesis nula.

A continuación, se realizan las comparaciones de los modelos en relación al modelo 3. Siguiendo el test Bonferroni-Dunn, la distancia crítica es:

$$DC = 2{,}394\sqrt{\frac{4\cdot 5}{6\cdot 15}} = 1{,}13$$

El valor crítico de esta prueba es menos restrictivo que la prueba de Nemenyi. Sin embargo, no se añaden nuevas hipótesis que se puedan rechazar.

A continuación, se comparan los modelos con el modelo M3 para contrastar la hipótesis de si este modelo da resultados significativamente distintos a los otros tres. Para realizar los cálculos de *z*, se calcula el error típico:

$$ET = \sqrt{\frac{4\cdot 5}{6\cdot 15}} = 0{,}47$$

La tabla siguiente contiene los cálculos necesarios:

Tabla 11. Resultados del contraste de hipótesis según el test de Holm

	R_3-R_j	Z	Valor p	i	α / (K-i)	H0
M3-M1	1,60 - 3,20 = -1,60	-3,40	0,0006738	1	0,017	Rechazo H0
M3-M2	1,60 - 2,27 = -0,67	-1,42	0,1556077	3	0,050	No se rechaza H0
M3-M4	1,60 - 2,93 = -1,33	-2,82	0,0048024	2	0,025	Rechazo H0

Según el método de Holm, se inician los contrastes con el valor de *p* más significativo: p_1. Se compara $p_1 < \alpha / 3$ que corresponde a 0,00067 < 0,017. Se rechaza la hipótesis nula de que M3 y M1 son equivalentes. A continuación, se compara $p_2 < \alpha / 2$, que corresponde a 0,0048 < 0,025. Se rechaza también la hipótesis nula que M3 y M4 dan resultados equivalentes. Por último, se compara M3 con M2: $p_3 <? \alpha$ con los valores correspondientes: $0{,}1556 <? 0{,}05$. Como la condición no se cumple, no se puede rechazar la hipótesis nula de que los métodos M3 y M2 son equivalentes.

El método de Hochberg empieza el contraste de hipótesis con el valor mayor de p. En este caso, compara $p_3 <^{?} \alpha$ que no se cumple $(0{,}155 > 0{,}05)$. No se rechaza la hipótesis nula. A continuación, se contrasta el valor de p_2 con $\alpha / 2$. Como ahora se cumple la condición (0,0048 < 0,025), se rechaza la hipótesis y por tanto, se rechazan las demás hipótesis, concluyendo que M3 es significativamente distinto a M1 y M4, respectivamente.

Los resultados de la comparación múltiple según Nemenyi han arrojado los mismos resultados que con el contraste usando las pruebas de Bonferroni-Dunn, Holm y Hochberg en relación al modelo M3. No necesariamente es así en general. Las pruebas que implican comparaciones múltiples son más conservadoras que las que comparan un modelo en relación a los demás. En este ejemplo, el contraste según Bonferroni-Dunn, Holm y Hochberg no ha desvelado nuevas diferencias significativas, lo cual refleja una diferencia significativa entre M3 y M4 y entre M3 y M1, mientras que M3 es bastante similar a M2 en los resultados obtenidos.

4.2.6. Reflexión sobre la comparación múltiple de clasificadores

En este apartado se han tratado las pruebas más habituales en la comparación de resultados de modelos de análisis de datos. Sin embargo, algunos autores van más allá con la propuesta de pruebas más adecuadas. El estudiante puede consultar los trabajos de Salzberg (1997) y García y Herrera (2008) para más detalles.

Resumen

En este módulo se ha tratado el diseño experimental de los métodos de analítica de datos. El primer paso consiste en hacer uso de una metodología de analítica de datos que especifique los pasos a seguir en función del objetivo del análisis. Este objetivo fijará asimismo el tipo de modelo a extraer de los datos (siendo los más comunes los modelos de clasificación, regresión, asociación y agrupación), lo cual determinará la elección de un conjunto de algoritmos de analítica posibles. Se han revisado las principales métricas de evaluación de la calidad del modelo como la precisión en modelos de clasificación o la raíz del error cuadrático medio para problemas de regresión. La evaluación del modelo consiste en estimar la precisión (u otra métrica adecuada) real del modelo, es decir, la precisión del modelo cuando se aplica a la población de interés. Habitualmente la estimación se realiza sobre una muestra pequeña de esta población y por ello son necesarios los métodos de remuestreo de la muestra de datos original.

Como es bien sabido que no existe un algoritmo óptimo para todo tipo de problemas ni existe una configuración óptima predeterminada de un algoritmo, es necesario estimar el mejor modelo para el problema dado. Para ello, se ha profundizado en la estimación de la calidad del modelo y la comparación de modelos. Se han distinguido varios escenarios, según si se comparan dos modelos o múltiples, sobre un mismo problema o en múltiples problemas y mediante pruebas paramétricas o no paramétricas.

El diseño experimental aplicado a la obtención de resultados de modelos de análisis de datos es análogo al diseño experimental que se puede aplicar a otros ámbitos. Así, la comparación entre los resultados de dos modelos puede equipararse a la respuesta fisiológica de dos tratamientos médicos o al rendimiento de un grupo de estudiantes antes y después de la enseñanza de un nuevo método de estudio. La complejidad de la evaluación de los resultados de los modelos de análisis de datos reside en la dificultad de estimar la precisión del modelo, y la minimización del sesgo y varianza mediante métodos de remuestreo. Por lo demás, los métodos de validación estadística tratados en este módulo pueden aplicarse a cualquier otro ámbito que cumpla las mismas condiciones.

Bibliografía

Cohen, J. (1960). «A coefficient of agreement for nominal scales». *Educational and Psychological Measurement* (n.º 20, págs. 37–46).

Davies, D. L.; Bouldin, D. W. (1979). «A cluster separation measure». *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1* (n.º 2, págs. 224–227). Disponible en: <10.1109/TPAMI.1979.4766909>.

Demsar, J. (2006). «Statistical Comparisons of Classifiers over Multiple Data Sets». *Journal of Machine Learning Research* (n.º 7, págs. 1-30).

Dietterich, T. G. (1998). «Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms». *Neural Computation* (n.º 10 (7), págs. 1895-1923).

García, S.; Herrera, F. (2008). «An Extension on "Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons». *Journal of Machine Learning Research* (n.º 9, págs. 2677-2694).

Jain, P.; Sharma, P. (2015). *Behind Every Good Decision: How Anyone Can Use Business Analytics to Turn Data into Profitable Insight*. NY: Amacom.

Kuzon, W.; Urbancheck, M.; McCabe, J. (1996). «The Seven Deadly Sins of Statistical Analysis». *Annals of Plastic Surgery* (págs. 265-272).

Lim, T.; Loh, W.; Shih, Y. (2000). «A Comparison of Prediction Accuracy, Complexity, and Training Time of Thirty-three Old and New Classification Algorithms». *Machine Learning* (n.º 40, págs. 203–229).

Lichman, M. (2013). *UCI Machine Learning Repository*. Irvine, CA: University of California, School of Information and Computer Science. Disponible en: <http://archive.ics.uci.edu/m1>.

Salzberg, S. L. (1997). «On Comparing Classifiers: Pitfalls to Avoid and a Recommended Approach». *Data Mining and Knowledge Discovery* (n.º 1, págs. 317–327).

Stapor, K. (2017). «Evaluating and Comparing Classifiers: Review, Some Recommendations and Limitations». En: Kurzynski, M.; Wozniak, M.; Burduk, R. (eds.). «Proceedings of the 10th International Conference on Computer Recognition Systems CORES 2017». *Advances in Intelligent Systems and Computing* (n.º 578). Cham (Suiza): Springer.

Vanwinckelen, G.; Blockeel, H. (2012). «On Estimating Model Accuracy with Repeated Cross-Validation». *21st Belgian-Dutch Conference on Machine Learning* (págs. 39-44).

Witten, I. H.; Frank, E.; Hall, M. A. (2011). *Data Mining. Practical Machine Learning Tools and Techniques* (3ª edición). Burlington: Morgan Kaufmann.

Zumel, N.; Mount, J. (2014). *Practical Data Science with R*. Shelter Island (Nueva York): Manning.