Minería de datos

MÁSTER UNIVERSITARIO EN INVESTIGACIÓN EN INTELIGENCIA ARTIFICIAL

Práctica 1: Selección de modelos

Jon Etxeberria San Millán

jecheverr33@alumno.uned.es

Curso: 2021-2022

Práctica 01

Contenido

1. Breve resumen teórico del problema de la selección de modelos

El concepto de Aprendizaje Automático o Machine Learning consiste en proveer a las máquinas o computadoras de la capacidad de resolver problemas nuevos en base a una experiencia de datos adquirida de forma automática[1] Una definición orientada a la ingeniería según Tom Mitchell en 1997 [2]: *"Se dice que un programa de computadora aprende de la experiencia E con respecto a alguna tarea T y alguna medida de desempeño P, si su desempeño en T, medido por P, mejora con experiencia E".*

El Machine Learning[3], básicamente supone la predicción de un evento o clasificación según unos datos históricos con los que se predice un valor numérico o categórico (de clasificación). Tipos principales de ML:

Tabla 1:

TIPO DE ML	CARACTERÍSTICAS PRINCIPALES
APRENDIZAJE SUPERVISADO	Datos de entrada que han sido previamente etiquetados: Clasificación y Regresión.
APRENDIZAJE NO SUPERVISADO	Los datos de entrada no están etiquetados.
APRENDIZAJE POR REFUERZO	Este tipo de Aprendizaje usa el modelo prueba y error, en una especie de aprendizaje que se basa en la recompensa de los comportamientos deseados y penalización de los no deseados.
APRENDIZAJE SEMISUPERVISADO	Se trata de un tipo de Aprendizaje intermedio entre el Supervisado y el No supervisado donde destacan

1.1 Definición de selección de modelos

La selección de modelos es el proceso de elegir uno entre muchos modelos candidatos para un problema de modelado predictivo. La primera pregunta que un científico de datos se formula para empezar a trabajar con un dataset es el tipo de algoritmo de Machine Learning apropiado para el análisis de los datos disponibles. El modelo para elegir depende en gran medida i) objetivo a obtener con los datos y la naturaleza de estos ii) requisitos de exigencia respecto a diferentes parámetros (precisión deseada, tiempo de entrenamiento…)

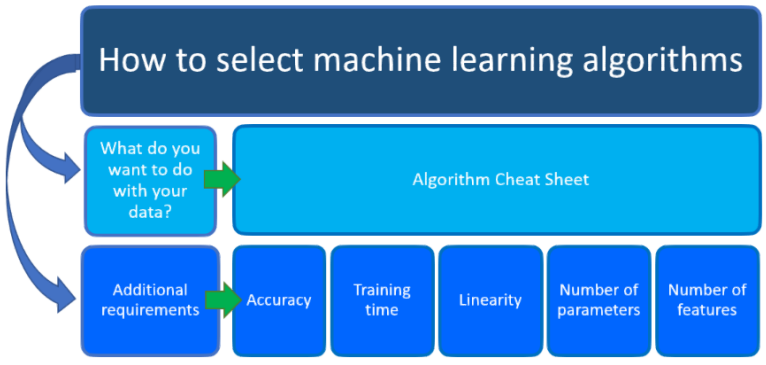

Ilustración 1: Variables a tener en cuenta para la selección de modelos (obtenido de https://docs.microsoft.com/es-es/azure/machine-learning/how-to-select-algorithms)

En definitiva, la selección de modelos puede definirse como : "*La selección de modelos es el proceso de seleccionar un modelo de aprendizaje automático final de entre una colección de modelos de candidatos para un conjunto de datos de entrenamiento*"

1.2 Modelos analizados en la practica

Tal y como se propone en el enunciado, la comparación entre modelos se realiza entre los algoritmos: K-NN, Naive Bayes y Regresión logística. Estos son modelos adecuados a el dataset Iris que se trata de un problema de clasificación.

Tabla 2: Clasificadores principales analizados en la práctica

CLASIFICADOR	CARACTERÍSTICAS
KNN	El ***algoritmo K-NN*** es un algoritmo que pertenece al Aprendizaje Supervisado, por lo que los datos de entrada están etiquetados y se conocen su salida. Se trata de datos de entrada formado por varios atributos descriptivo, y una clase de salida. Se trata de un algoritmo no paramétrico. Este algoritmo mantiene todos los datos con los que realiza el aprendizaje, al contrario que otros modelos. El algoritmo cuando parece una entrada nueva no vista encuentra los k ejemplos más cercanos devolviendo la etiqueta mayoritaria en caso de regresión, o la etiqueta promedio en caso de clasificación
REGRESIÓN LOGÍSTICA	Modelo de clasificación linear que busca conocer la relación entre: i) **variable dependiente cualitativa**, dicotómica (regresión logística binaria) o con más de dos categorías (regresión logística multinomial) ii) Una o más variables explicativas independientes, llamadas **covariables**, ya sean cualitativas o cuantitativas. Este modelo tiene 3 objetivos principales: i) cuantificar la importancia de la relación existente entre cada una de las covariables y la variable dependiente ii) clarificar la existencia de interacción y confusión entre covariables respecto a la variable dependiente iii) clasificar individuos dentro de las categorías (presente/ausente) de la variable dependiente.
NAIVE BAYES	Naïve Bayes o el Ingenuo Bayes es uno de los algoritmos más simples y poderosos para la clasificación basado en el Teorema de Bayes y que asume que el efecto de una característica particular en una clase es independiente de otras características.

Hay que comentar que para la realización de la práctica han sido utilizados más clasificadores para poder aprender el uso de los modelos y realizar comparativas extras que pudieran ofrecer conclusiones más robustas a las obtenidas sólo por 3 clasificadores.

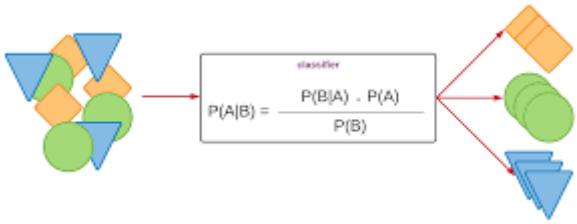

Ilustración 2: Algoritmo de BayesNet (obtenido de https://hands-on.cloud/implementing-naive-bayes-classification-using-python/)

1.3 Técnicas de selección de modelos

Para poder aplicar el Machine Learning (ML) lo primero sería conveniente disponer de datos "suficientes", que pudieran ser casi infinitos según la complejidad del problema. Medidas probabilísticas: se selecciona un modelo a través del error y la complejidad de la muestra. Analizamos las siguientes técnicas:

1.3.1 Técnicas estadísticas

La comparación entre los resultados obtenidos por diferentes modelos, pueden aplicarse métodos estadísticos que den validez a los resultados individuales obtenidos por cada algoritmo. Cuando se obtienen resultados de dos o más modelos de ML (aplicados sobre mismos o distintos conjuntos de datos) y se pretende compararlos formalmente para determinar cuál se comporta mejor ante un determinado problema, no es formalmente válido escoger aquel que mejor resultado o métrica haya dado. Hay que demostrar que las diferencias son estadísticamente

significativas, y no debidas al azar o ruido en los datos, de lo contrario no podemos asumir que los modelos en cuestión se comportan de forma distinta.

Las métricas de desempeño son varias([4], [5], [6]), aunque en función del tipo de algoritmo conviene utilizar unas frente a otras(. Para la clasificación las principales son i) Accuracy: Puede definirse como el porcentaje de predicciones correctas hechas por el modelo de clasificación. ii) Precision: Indica, de todas las predicciones positivas, cuántas son realmente positivas iii) TPR/Sensitivity/Recall: Indica, de todos los valores realmente positivos, cuántos se predicen como positivos. iv) Specificity: Indica, de todos los valores realmente negativos, cuántos se predicen como negativos.

Una vez analizadas las métricas habituales de medición de los algoritmos de forma individual, para la comparación entre modelos hemos dicho que existen test específicos estadísticos que haciendo uso de la estadística dan un resultado formal sobre cuál es el que mejor resultado ha tenido. Desde el punto de vista estadístico, se establece una hipótesis nula a refutar:

H0: los modelos presentan el mismo rendimiento, dicho de otra forma, las métricas escogidas para evaluar su desempeño son iguales.

Existen diferentes requisitos para poder aplicar según que técnica estadística, pudiendo tratarse de tests paramétricos o no paramétricos.

1. Test paramétricos: Las pruebas paramétricas están basadas en la ley de distribución de la variable que se estudia. Los datos deben distribuirse de forma "normalizada": la media y la desviación estándar. Además, debe cumplir con i) Homocedasticidad: los grupos deben presentar variables uniformes, es decir homogéneas ii) Errores: los errores que surjan deben ser independientes.
2. Test no paramétricos: En este caso los datos no siguen ningún tipo de distribución conocida, siendo imposible definir la misma a priori.

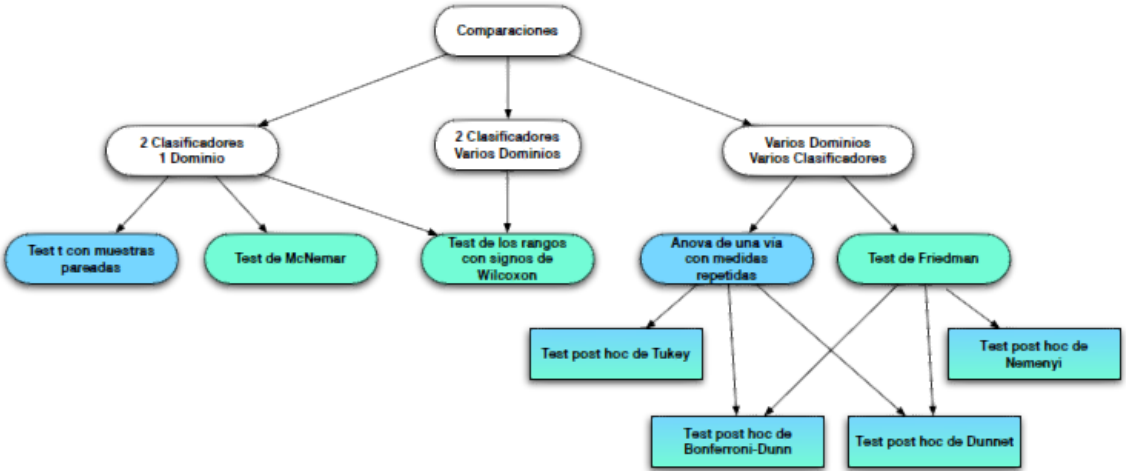

Ilustración 3: Tipo de test aplicable para cada problema (obtenido de https://rpubs.com/Cristina_Gil/comparacion_estadistica_modelos)

1.3.1.1 Prueba de t-student con muestras pareadas

Las pruebas t de dos muestras [7] son pruebas estadísticas que se utilizan para comparar las medias de dos poblaciones. También conocidas como pruebas t de Student, sus resultados se utilizan para determinar si existe una diferencia significativa entre la media de dos muestras que es poco probable que debido a un error de muestreo o al azar.

Una prueba t pareada también conocida como prueba t dependiente o correlacionada es una prueba estadística que compara los promedios / medias y las desviaciones estándar de dos grupos relacionados para determinar si hay una diferencia significativa entre los dos grupos.

- Se produce una diferencia significativa cuando es poco probable que las diferencias entre los grupos se deban a un error de muestreo o al azar.
- Los grupos pueden estar relacionados por ser el mismo grupo de individuos, el mismo artículo o estar sujetos a las mismas condiciones.

Las hipótesis de la prueba son:

1. El hipótesis nula H0 indica que no hay una diferencia significativa entre las medias de los dos grupos.
2. El hipótesis alternativa H1 establece que existe una diferencia significativa entre las dos medias poblacionales y que es poco probable que esta diferencia se deba a un error de muestreo o al azar.

Esta técnica se basa en los siguientes supuestos:

1. La variable dependiente se distribuye normalmente
2. Las observaciones se muestrean de forma independiente
3. La variable dependiente se mide en un nivel incremental, como proporciones o intervalos.
4. Las variables independientes deben constar de dos grupos relacionados o pares coincidentes.

A continuación, en la Función 2 se observa la fórmula del test, en donde d es la diferencia del valor pareado y n es el número de individuos.

$$t = \frac{\Sigma d}{\sqrt{\frac{n(\Sigma d^2)-(\Sigma d)^2}{n-1}}} \qquad \text{(Función 1)}$$

1.3.1.2 Test de McNemar

El test de McNemar[8] es la alternativa a los test $\chi 2$ de Pearson y al test exacto de Fisher cuando: los datos son pareados, se trata de tablas 2x2 y ambas variables son dicotómicas (binomiales). El test de McNemar estudia si la probabilidad de evento verdadero para una variable es igual en los dos niveles de otra variable. Requiere de las siguientes condiciones:

- Se trata de datos pareados.
- Se estudian dos variables, ambas de tipo binomial (dicotómicas). Tabla 2x2.
- La suma de eventos que pasan de positivo a negativo y de negativo a positivo ha de ser > 25, de lo contrario se emplea un test binomial en el que el número de aciertos es el número de eventos que han pasado de positivos a negativos y el número total de intentos es la suma de todos los que han cambiado (de positivos a negativos y de negativos a positivos).

Para entender el funcionamiento, supóngase que un grupo de sujetos pasa un test cuyo resultado es binomial (positivo y negativo) antes y después de un tratamiento. El objetivo de la prueba es determinar si el tratamiento hace cambiar los test de positivo a negativo o viceversa.

Tabla 3: Tabla de McNemar

..	Después Positivo	Después Negativo	Total
Antes Positivo	a	b	a+b
Antes Negativo	c	d	c+d
Total	a+c	b+d	n=a+b+c+d

Si el valor de una variable es independiente de la otra, se esperaría que las proporciones de pasar negativo a positivo fuesen iguales a las de pasar de positivo a negativo. Esto significa que pc+pd=pb+pd y pc+pd=pb+pd; lo que queda como pb=pc (hipótesis nula).

H0: pb=pc

HA: pb≠pc

El estadístico del test de McNemar, siempre y cuando se cumpla la condición mínima de eventos, sigue una distribución $\chi 2$ con 1 grado de libertad.

$$\chi 2 = \frac{(b-(b+c)/2)^2}{(b+c)/2} + \frac{(c-(b+c)/2)^2}{(b+c)/2} = \frac{(b-c)^2}{b+c} \qquad \text{(Función 2)}$$

1.3.2 Métodos de remuestreo: se selecciona un modelo a través del error estimado fuera de la muestra.

Los métodos de remuestreo buscan estimar el desempeño de un modelo (o más precisamente, el proceso de desarrollo del modelo) en datos que estén fuera de la muestra. Esto se logra dividiendo el conjunto de datos de entrenamiento en conjuntos de entrenamiento y de test, ajustando un modelo en el conjunto de entrenamiento y evaluándolo en el conjunto de test. Luego, este proceso puede repetirse varias veces y se informa el rendimiento medio en cada ensayo. Es un tipo de estimación de Monte Carlo del rendimiento del modelo en datos fuera de la muestra, aunque cada prueba no es estrictamente independiente ya que, según el método de remuestreo elegido, los mismos datos pueden aparecer varias veces en diferentes conjuntos de datos de entrenamiento o conjuntos de datos de prueba. Métodos comunes de remuestro son: Random train/test split ii) Cross Validation (K-fold, LOOCV…) iii) Bootstrap

1.3.2.1 Random train/test split

El método más sencillo de validación consiste en repartir aleatoriamente las observaciones disponibles en dos grupos, uno se emplea para entrenar al modelo y otro para evaluarlo. Si bien es la opción más simple, tiene dos problemas importantes:

La estimación del error es altamente variable dependiendo de qué observaciones se incluyan como conjunto de entrenamiento y cuáles como conjunto de validación (problema de varianza).

Al excluir parte de las observaciones disponibles como datos de entrenamiento (generalmente el 20%), se dispone de menos información con la que entrenar el modelo y, por lo tanto, se reduce su capacidad. Esto suele tener como consecuencia una sobrestimación del error comparado al que se obtendría si se emplearan todas las observaciones para el entrenamiento (problema de bias).

1.3.2.2 Leave One Out Cross-Validation (LOOCV)

El método LOOCV en un método iterativo que se inicia empleando como conjunto de entrenamiento todas las observaciones disponibles excepto una, que se excluye para emplearla como validación. Si se emplea una única observación para calcular el error, este varía mucho dependiendo de qué observación se haya seleccionado. Para evitarlo, el proceso se repite tantas veces como observaciones disponibles, excluyendo en cada iteración una observación distinta, ajustando el modelo con el resto y calculando el error con dicha observación. Finalmente, el error estimado por el LOOCV es el promedio de todos lo i errores calculados.

El método LOOCV permite reducir la variabilidad que se origina si se divide aleatoriamente las observaciones únicamente en dos grupos. Esto es así porque al final del proceso de LOOCV se acaban empleando todos los datos disponibles tanto como entrenamiento como validación. Al no haber una separación aleatoria de los datos, los resultados de LOOCV son totalmente reproducibles.

La principal desventaja de este método es su coste computacional. El proceso requiere que el modelo sea reajustado y validado tantas veces como observaciones disponibles (n) lo que en algunos casos puede ser muy complicado. Excepcionalmente, en la regresión por mínimos cuadrados y regresión polinomial, por sus características matemáticas, solo es necesario un ajuste, lo que agiliza mucho el proceso.

LOOCV es un método de validación muy extendido ya que puede aplicarse para evaluar cualquier tipo de modelo. Sin embargo, determinados autores consideran que, al emplearse todas las observaciones como entrenamiento, se puede estar cayendo en overfitting, por lo que, aun considerándolo muy aceptable, recomiendan emplear K-Fold Cross-Validation.

1.3.2.3 K-Fold Cross-Validation

El método K-Fold Cross-Validation [9] es también un proceso iterativo. Consiste en dividir los datos de forma aleatoria en k grupos de aproximadamente el mismo tamaño, k-1 grupos se emplean para entrenar el modelo y uno de los grupos se emplea como validación. Este proceso se repite k veces utilizando un grupo distinto como validación en cada iteración. El proceso genera k estimaciones del error cuyo promedio se emplea como estimación final. Dos ventajas del método K-Fold Cross-Validation frente al LOOCV:

- Requerimientos computacionales: el número de iteraciones necesarias viene determinado por el valor k escogido. Por lo general, se recomienda un k entre 5 y 10. LOOCV es un caso particular de K-Fold Cross-Validation en el que k = nº observaciones, si el data set es muy grande o el modelo muy complejo, se requiere muchas más iteraciones.
- Balance entre bias y varianza: la principal ventaja de K-fold CV es que consigue una estimación precisa del error de test gracias a un mejor balance entre bias y varianza. LOOCV emplea n-1 observaciones para entrenar el modelo, lo que es prácticamente todo el set de datos disponible, maximizando así el ajuste del modelo a los datos disponibles y reduciendo el bias. Sin embargo, para la estimación final del error se promedian las estimaciones de n modelos entrenados con prácticamente los mismos datos (solo hay un dato de diferencia entre cada conjunto de entrenamiento), por lo que están altamente correlacionados. Esto se traduce en un mayor riesgo de overfitting y por lo tanto de varianza. En el método K-fold CV los k grupos empleados como entrenamiento son mucho menos solapantes, lo que se traduce en menor varianza al promediar las estimaciones de error.

Aunque emplea menos observaciones como entrenamiento que LOOCV, son un número suficiente como para no tener un bias excesivo, por lo que el método K-fold CV con valores de k= [5, 10] consigue un mejor balance final.

1.3.2.4 Repeated k-Fold-Cross-Validation

Es exactamente igual al método k-Fold-Cross-Validation, pero repitiendo el proceso completo n veces. Por ejemplo, 10-Fold-Cross-Validation con 5 repeticiones implica a un total de 50 iteraciones ajuste-validación, pero no equivale a un 50-Fold-Cross-Validation.

1.3.2.5 Bootstrapping

Una muestra bootstrap es una muestra obtenida a partir de la muestra original por muestreo aleatorio con reposición, y del mismo tamaño que la muestra original. Muestreo aleatorio con reposición (resampling with replacement) significa que, después de que una observación sea extraída, se vuelve a poner a disposición para las siguientes extracciones. Como resultado de este tipo de muestreo, algunas observaciones aparecerán múltiples veces en la muestra bootstrap y otras ninguna. Las observaciones no seleccionadas reciben el nombre de out-of-bag (OOB). Por cada iteración de bootstrapping se genera una nueva muestra bootstrap, se ajusta el modelo con ella y se evalúa con las observaciones out-of-bag.

- Obtener una nueva muestra del mismo tamaño que la muestra original mediante muestro aleatorio con reposición.
- Ajustar el modelo empleando la nueva muestra generada en el paso 1.
- Calcular el error del modelo empleando aquellas observaciones de la muestra original que no se han incluido en la nueva muestra. A este error se le conoce como error de validación.
- Repetir el proceso n veces y calcular la media de los n errores de validación.
- Finalmente, y tras las n repeticiones, se ajusta el modelo final empleando todas las observaciones de entrenamiento originales.

La naturaleza del proceso de bootstrapping genera cierto bias en las estimaciones que puede ser problemático cuando el conjunto de entrenamiento es pequeño. Existen ciertas modificaciones del algoritmo original para corregir este problema, algunos de ellos son: 632 method y 632+ method.

1.4 Calidad de los resultados obtenidos con el experimento

Existen tres métodos principales[10] para juzgar la calidad de un experimento:

1. El error ***Tipo I*** es la probabilidad de que la conclusión de un experimento sea que haya una diferencia entre algoritmos, mientras que en realidad no los hay. En teoría, el error tipo I es igual al "nivel de significación" elegido para el prueba de hipótesis si ninguno de los supuestos de la prueba son violados. En la práctica, la suposición de independencia a menudo se viola, lo que da como resultado un Tipo I elevado error.
2. El error ***Tipo II*** es la probabilidad de que la conclusión de un experimento sea que no hay diferencia entre los algoritmos, mientras que en realidad la hay. El *poder* se define como: "1 - el error de tipo II". El poder no es directamente controlable como lo es el error Tipo I. Sin embargo, hay es un equilibrio entre la potencia y el error de tipo I y un Se puede obtener una mayor potencia a costa de una mayor Error tipo I. La relación exacta entre los dos depende del diseño experimental.
3. La ***replicabilidad*** de un experimento es una medida de cómo bien, el resultado de un experimento se puede reproducir.

2. Resumen de las técnicas, de sus puntos fuertes y débiles en comparación con otras alternativas de las expuestas en el punto anterior.

Tal y como se ha propuesto en el enunciado, los algoritmos principales que se van a utilizar van a ser: el modelo más básico que es el modelo (que no por ello el peor ni el que peores resultados obtiene) Logistic Regresion, el K-Nearest Neighbors y el Naive Bayes. Todos estos son los algoritmos propuestos en el enunciado. Además, también se trabajarán otros algoritmos para enriquecer la práctica y comparar resultados.

2.1 Características de los Algoritmos principales de la práctica

Las características más destacables de los algoritmos utilizados se representan en la siguiente Tabla 4.

Tabla 4: Características principales de los algoritmos comparados en la práctica

Modelo	Ventajas	Inconvenientes	Utilidad
Regresión Logística	• Fácil de entender y explicar. • Es rápido de modelar y es muy útil cuando la relación a modelar no es extremadamente compleja.	• No se puede modelar relaciones complejas. • No se pueden capturar relaciones no lineales sin transformar la entrada. • Puede sufrir con valores atípicos.	• Dar un primer vistazo a un conjunto de datos. • Cuando se tiene datos numéricos con muchas características. • Realizar predicciones econométricas.
KNN	• Alta precisión, insensible a valores atípicos y sin suposiciones de entrada de datos. • Sencillo, fácil de entender y muy potente	• Muy sensible a los atributos irrelevantes y la maldición de la dimensionalidad • Muy sensible al ruido • Lento, si hay muchos datos de entrenamiento(O(n*d) en almacenamiento y en tiempo) • Depende de que la función de distancia sea la adecuada	• KNN también puede manejar problemas de regresión, es decir, predicción • Aplicaciones en Física.
Naive Bayes	• Fácil de implementar y Rápido. • Requiere menos datos de entrenamiento. • Es altamente escalable. • Puede hacer predicciones probabilísticas. • Puede manejar datos continuos y discretos.	• Necesita calcular la probabilidad previa. • La alta tasa de error de las decisiones de clasificación. • Es muy sensible a la forma de expresión de los datos de entrada. • Usar la suposición de la independencia de los atributos de la muestra,	• Se utiliza mucho en NLP (Natural Language Processing) • Tareas más complejas como reconocer un idioma Detección de intrusiones o anomalías en redes informáticas.

2.2 Dataset

Se ha utilizado el dataset de Iris. Este dataset sólo contiene dos racimos, con una separación obvia y clara. Uno de los racimos contiene Iris setosa, mientras el otro contiene ambos Iris virgínica y Iris versicolor y no es separable, sino que uno tiene la información de especies usadas por Fisher. Esto hace que el conjunto de datos sea un buen ejemplo para explicar la diferencia entre las técnicas supervisadas y no supervisadas en la minería de datos: el modelo discriminante lineal de Fisher solo se puede obtener cuando se conocen las especies del objeto: las etiquetas de clase y los grupos no son necesariamente los mismos.

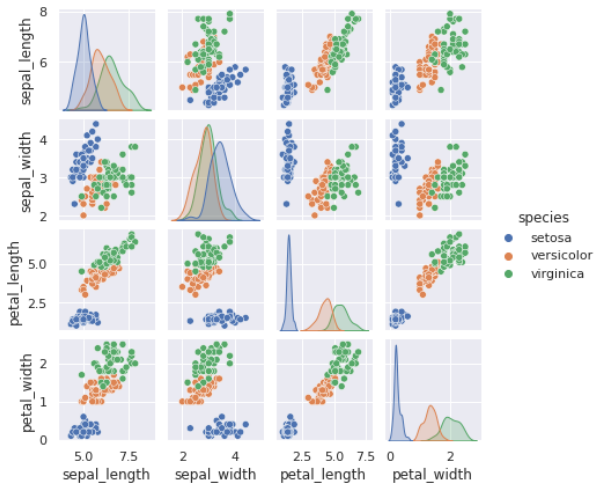

Ilustración 4: Distribución de las clases según los atributos de entrada [setosa(azul), versicolor (naranja), virgínica (verde)]

2.3 Test 5x2 t-muestras pareadas

El test t-Student Pareado se aplicará a los resultados de aplicar el K-fold Validation con k=5. Las ventajas que ofrece el 5x2 K-fold son la de menor coste computacional unido a consigue una estimación precisa del error de test gracias a un mejor balance entre bias y varianza. Combinando con t-Student asume que: i) las muestras se extraen de forma independiente y aleatoria ii) la distribución de los residuos entre los dos grupos debe seguir la distribución normal iii) las variaciones entre los dos grupos son iguales, baja tasa de falsos positivos. Las ventajas que ofrece esta técnica son i) simplicidad de interpretación ii) robusta iii) requiere de pocos datos iv) fácil de calcular. Dadas las características del dataset, ya que los datos tienen subconjuntos no independientes, pueden surgir pegas y errores de resultado.

2.4 Test de McNemar

El segundo test es el de McNemar. Tal y como se explicó en el punto anterior, el test de McNemar estudia si la probabilidad de evento verdadero para una variable es igual en los dos niveles de otra variable. Las condiciones para el uso de este test son i) Se trata de datos pareados ii) Se estudian dos variables, ambas de tipo binomial (dicotómicas). Ofrece la ventaja de que obtiene una baja tasa de falsos positivos y es rápido, ya que solo debe ejecutarse una vez. Es una buena opción para conjunto de datos relativamente grandes y/o el ajuste del modelo sólo puede hacerse una vez. La posibilidad de repetición de ajuste de los modelos, hace que el 5x2cv sea una buena opción porque considera el efecto de los subconjuntos de entrenamiento variados o remuestreados en el ajuste del modelo.

3. Descripción de los resultados obtenidos

El dataset iris cuenta con 4 variables independientes (sepal_length, sepal_width, petal_length, petal_width), y una clase (setosa, vesicolor, virginica). En código se han aplicado diferentes clasificadores, aunque los resultados finales son los que se piden de un clasificador lineal y otro basado en los vecinos más cercanos. El dataset se ha dividido en datos de entrenamiento (75%) y datos de test (25%).

Los resultados obtenidos para el entrenamiento de los algoritmos han sido los siguientes:

Tabla 5: Resultados de los algoritmos en entrenamiento

Nº	Algoritmo	Precisión
1	Logistic Regression	0.964394
2	KNN	0.955303
3	Support Vector Classifier	0.955303
4	NBGauss	0.955303
5	Decission Tree	0.946212
6	Random Forest	0.928030

Se observa que el algoritmo que mejor resultado ha dado en test ha sido Logistic Regression (96,4394%), a pesar de que se supone más simple.

A los algoritmos con mejor resultado, se ha realizado una búsqueda de mejores parámetros aplicando la función GridSearchCV de sklearn, mejorando el resultado a 98,1818%, con los siguientes parámetros: *{'C': 100, 'penalty': 'l2', 'solver': 'newton-cg'}.*

En referencia a los resultados obtenidos con los algoritmos seleccionados para la práctica en test:

Tabla 6: Resultados de los algoritmos utilizados con el dataset iris

Métrica	LR	NBGauss	KNN
Precision	97.37%	94,74%	97,37%
RMSE	0,16222142113076254	0,22941573387056177	0,16222142113076254
Matriz de Confusión	16 0 0 [0 8 0] 0 1 13	16 0 0 [0 8 0] 0 2 12	16 0 0 [0 7 1] 0 0 14

En principio de los 3 algoritmos aplicados, se puede observar que los que mejor resultado obtienen son el LR y el KNN, que obtienen una misma precisión (97,37%) aunque los errores como puede observarse en la matriz de confusión son distintos: LR tiene un error confundiendo una clase3(virgínica) con clase2(versicolor), mientras el KNN confunde una clase2(versicolor) con una clase3(virgínica). También se obtiene el error cuadrático medio de cada modelo, en donde tanto LR como KNN igualan (0,16222142113076254) mejorando al modelo Naive Bayes Gaussiano.

3.1 Grid Search

Se ha hecho una prueba de mejora de parámetros de los algoritmos que mejor resultado han ofrecido en entrenamiento. Se ha aplicado la función GridSearchCV de sklearn, mejorando el resultado a 98,1818%, con los siguientes parámetros: {'C': 100, 'penalty': 'l2', 'solver': 'newton-cg'}.

3.2 Comparación Algoritmo: Test de student pareado sobre 5x2-fold cross validation

Para esta prueba, como modelos seleccionados quedan LR y KNN que son los que mejor resultado han conseguido en training y en test. Así, se realiza la prueba estadística de selección de modelos para dilucidar el mejor. Para la realización de la comparación de los algoritmos, aplicando el test de student pareado, hay que usar exactamente los mismos datos tanto en

entrenamiento como en test. Se recomienda la aplicación de 5x2cv en vez de 10cv, que divide los datos en ½ para entrenamiento y ½ para test, 5 veces.

Se trata de una prueba estadística que compara los promedios / medias y las desviaciones estándar de dos grupos relacionados para determinar si hay una diferencia significativa entre los dos grupos. Trabaja con 2 hipótesis i) hipótesis nula "H0" indica que no hay una diferencia significativa entre las medias de los dos grupos ii) hipótesis alternativa H1 establece que existe una diferencia significativa entre las dos medias poblacionales y que es poco probable que esta diferencia se deba a un error de muestreo o al azar.

3.2.1 Resultados comparación LR-KNN

Al realizar la comparación entre los algoritmos Logistic Regression y KNN, obtenemos un valor t= |-717| y p = 0,505. Suponiendo un α = 0,05, se puede comprobar que p > 0,05, lo que hace que no se pueda descartar la hipótesis nula. Esto significa que los dos algoritmos comparados no obtienen un rendimiento significativamente diferente en su resultado obtenido.

3.3 Test de McNemar

La comparativa de los modelos anteriores haciendo uso del test de Student pareado, no ha permitido obtener conclusiones sobre el rendimiento de los algoritmos utilizados, por lo que aplicaremos el test de McNemar, que centra el interés en comparar si hay o no cambios significativos entre las mediciones efectuadas en dos momentos diferentes, teniendo otra medida de comparación que permita obtener conclusiones satisfactorias sobre la diferencia de rendimiento de los algoritmos utilizados.

Para el test de McNemar se han comparado los algoritmos de LR con KNN y LR con NB, para ver si obteníamos alguna diferencia significativa en la comparativa, ya que la precisión del algoritmo de LR y KNN había sido la misma (97,37%). El nivel de significancia a utilizar es el mismo que en el caso anterior de α = 0,05.

3.3.1 Resultados comparación LR-KNN

Los datos obtenidos del test entre los modelos LR-KNN son $\chi 2$ = 1 y p = 1. En este caso comparativo, se observa que el valor de p es mayor que el nivel de significancia de α = 0,05.

Tabla 7: Aplicación del test de McNemar a la comparación de algoritmos LR-KNN

		LR	
		Correcto	Incorrecto
KNN	Correcto	36	1
	Incorrecto	1	0

El valor de p es superior al umbral de significancia predefinido, por lo que no se puede desestimar la hipótesis nula.

4. Conclusiones

La comparación realizada tanto con el test de McNemar como el de t-Student, no permiten descartar la hipótesis nula, haciendo que los resultados obtenidos con los diferentes clasificadores (KNN y Regresión Logística) sean significativamente diferenciados. Para este ejercicio hubiera sido adecuado ampliar el número de clasificadores y aplicar un test de ANOVA, para poder determinar qué clasificadores tienen mejor rendimiento. Hay que tener en cuenta que el conjunto iris sólo dispone de 150 elementos, por lo que es difícil que exista una gran diferencia entre los algoritmos para un dataset tan pequeño.

Aunque los test no han permitido obtener conclusiones sobre los datos, la distribución gráfica visual de los mismos permite observar cómo en general adquieren una distribución gaussiana similar las 3 clases, salvo la clase setosa que se desmarca del resto. Esto hace que los errores que han cometido los modelos sean precisamente en equivocarse entre las clase2(versicolor) y clase3(virgínica), ya que son dos clases indivisibles.

Existe una modificación al test t pareado, con la variable F denominado 5x2cv-f. Esta fue una modificación propuesta por Dietrich para resolver la dependencia del orden en el que se realizan los experimentos. Este nuevo estadístico sigue una distribución $F_{10,5,}$ haciendo el test más potente en determinadas circunstancias.

En próximas prácticas será recomendable hacer uso de más estadísticos que puedan obtenerse conclusiones claras para la selección del mejor algoritmo. Sin embargo, con los datos obtenidos, y tomando en cuenta los resultados particulares de ambos, ya que la prueba t y el test de McNemar no permiten obtener conclusiones definitivas, se puede decir que de forma intuitiva el Logistic Regression ofrece un ligero mejor rendimiento en las pruebas individuales realizadas.

También sería interesante aplicar las nuevas técnicas de las librería de scikit-learn en donde se combinan los algoritmos creando lo que se denomina Ensemble Learning. En general, el aprendizaje conjunto es muy poderoso y puede usarse no solo para problemas de clasificación sino también para regresión. Se trata de un modelo que hace predicciones basadas en varios modelos diferentes. Al combinar modelos individuales, el modelo de conjunto tiende a ser más flexible(menos sesgo) y menos sensible a los datos (menos variación).

5. Referencias

[1] D. Borrajo, J. González Boticario, and P. Isasi Viñuela, “Aprendizaje automático,” 2006.

[2] A. Géron, *Hands-on Machine Learning whith Scikit-Learing, Keras and Tensorfow*. 2019.

[3] H. Sciences, *Pattern Recognition and Machine Learning*, vol. 4, no. 1. Singapore: Springer Science+Business Media, LLC, 2016.

[4] T. Hastie, R. Tibshirani, and J. Friedman, “Statistics The Elements of Statistical Learning,” *Math. Intell.*, vol. 27, no. 2, pp. 83–85, 2009, [Online]. Available: http://www.springerlink.com/index/D7X7KX6772HQ2135.pdf.

[5] L.-P. Chen, *Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python*, vol. 63, no. 2. 2021.

[6] M. Peter, D. A. Aldo, F. Cheng, and S. Ong, “MATHEMATICS FOR MACHINE LEARNING,” Accessed: Mar. 21, 2022. [Online]. Available: https://mml-book.com.

[7] T. G. Dietterich, “Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms,” *Neural Comput.*, vol. 10, no. 7, pp. 1895–1923, Oct. 1998, doi: 10.1162/089976698300017197.

[8] Q. McNemar, “Note on the sampling error of the difference between correlated proportions or percentages,” *Psychom. 1947 122*, vol. 12, no. 2, pp. 153–157, Jun. 1947, doi: 10.1007/BF02295996.

[9] T. Fushiki, “Estimation of prediction error by using K-fold cross-validation,” *Stat. Comput. 2009 212*, vol. 21, no. 2, pp. 137–146, Oct. 2009, doi: 10.1007/S11222-009-9153-8.

[10] R. R. Bouckaert, “Estimating replicability of classifier learning experiments,” *Proceedings, Twenty-First Int. Conf. Mach. Learn. ICML 2004*, pp. 113–120, 2004, doi: 10.1145/1015330.1015338.