UNIVERSITAT POLITÈCNICA DE VALÈNCIA

TRABAJO FIN DE MASTER EN ANÁLISIS DE DATOS, MEJORA DE PROCESOS Y TOMA DE DECISIONES

# APLICACIÓN DE MÉTODOS ESTADÍSTICOS MULTIVARIANTES PARA LA EVALUACIÓN DE LA CALIDAD EN JAMONES, POR MEDIO DE ESTRUCTURAS DE DATOS N-DIMENSIONALES

AUTOR: Raquel Trecet Pizarro

TUTOR: José Manuel Prats Montalbán

CO-TUTOR: José Vicente García Pérez

Curso académico 2017-2018

-Aplicación de métodos estadísticos multivariantes para evaluación de la calidad en jamones, por medio de estructuras N-dimensionales-

# AGRADECIMIENTOS:

En primer lugar, quiero dar gracias a mi profesor José Manuel Prats Montalbán. Gracias por crear tan buen clima de trabajo y sobre todo por la gran ayuda que me has brindado durante estos dos años. Gracias por enseñarme esta ciencia con tantas aplicaciones.

En el ámbito personal, quiero dar las gracias a mi familia por darme la oportunidad de estudiar todo lo que he querido. Gracias por el inmenso apoyo que he tenido durante todos mis años de estudio. Por último, gracias a mi gran amor y compañero de todos estos años, Germán.

# ÍNDICE DE CONTENIDOS

# INTRODUCCIÓN

# 1.INTRODUCCIÓN

## 1.1 Proceso y calidad del jamón

España es uno de los principales productores del jamón ibérico. Numerosos países y personas consumen este alimento de gran aporte proteico y sabor. Un aspecto muy importante es que la calidad del producto sea buena. La calidad depende de muchos factores ambientales y biológicos del animal. Sin embargo, un factor importante en la calidad final es el proceso de curación para numerosas características texturales.

El jamón pasa por diferentes fases de curación que serán determinantes para su calidad final. El proceso se puede dividir en 3 grandes fases: salación, reposo y secado. El proceso de salación tiene como objetivo la estabilidad microbiana. Durante esta fase, se reduce la actividad del agua de la carne aportando las características sensoriales necesarias del producto. La etapa de reposo se efectúa a 4 ºC, la cual depende del jamón (es decir, en función del jamón varía el tiempo de reposo). Por último, en la etapa de secado o maduración se produce la deshidratación del jamón para alcanzar la estabilidad final del producto.

La calidad de un jamón (Ruiz, 2015) depende de factores antemorten y postmortem, es decir, variables que influyen sobre el estado final. Además, se desea demostrar cuáles son las que influyen significativamente sobre la calidad del jamón.

Existen muchos tipos de defectos en la calidad que pueden ser atribuidos a variables relacionadas con el aspecto, textura o sabor. *La pastosidad* es uno de los principales defectos en el sector del jamón curado, una característica de baja calidad ocasionada por un fenómeno desconocido. Conceptualmente es la falta de elasticidad del producto que evita que pueda volver a su estado original (García-Garrido, Quiles-Zafra, Tapiador, & Luque de Castro, 1999). Esta problemática ocasiona problemas de calidad graves, no puede controlarse durante el proceso de curación, sino que se detecta al final del proceso mediante análisis destructivos. Estos análisis destructivos consisten en el criterio de jueces expertos, que determinan si el jamón presenta esta característica. Los análisis se basan en la experiencia y entrenamiento que tienen los jueces en detectar los parámetros de calidad de un jamón: la textura, el sabor, la sal y otros muchos más. Una de las características en las que se basan los ensayos es en la clasificación del grado de pastosidad que presente un jamón para clasificarlo en: alto pastoso ("*High pastiness),* medio pastoso ("*Medium pastiness*") y bajo pastoso ("*Low pastiness*").

Este tipo de análisis presenta muchos inconvenientes, ya que cuenta con la gran inestabilidad de posibles errores humanos, criterios subjetivos y además no puede automatizarse el control y utilizarse en líneas de producción (problemas ligados a los conceptos de repetibilidad y reproducibilidad). Por ello, surge la necesidad de utilizar técnicas no destructivas durante el proceso del jamón, y emplear modelos predictivos y de análisis que puedan asegurarnos una calidad final del producto sin pastosidad.

Numerosos autores han tratado de estudiar el fenómeno de la pastosidad en los jamones. Sin embargo, en este trabajo se pretende estudiar las variables que afectan en la pastosidad así como las relaciones existentes entre ellas a partir de técnicas basadas en estructuras latentes. También, se realizarán varios modelos de clasificación basados en técnicas de aprendizaje supervisado que se evaluarán mediante la tasa de aciertos. De esta forma, a partir de clasificadores se podrá determinar si un jamón es pastoso en función de sus parámetros más importantes. Además, con estos modelos, se podrán eliminar variables que actualmente se están midiendo, ahorrando tiempo y disminuyendo costes económicos ligados a las mediciones.

Se va a trabajar con variables de distinta naturaleza. Las variables de proceso han sido registradas a través de un equipo ultrasonidos formado por dos placas (receptora y emisora). Este equipo midió las mismas variables durante todas las fases del jamón. Este método no requiere ninguna destrucción de la muestra, sino que requiere el jamón entero.

Una vez terminada la curación, se realizaron análisis destructivos (lonchas de jamón) para tomar medidas sensoriales, estáticas, bioquímicas y texturales (variables de calidad) obteniendo así la variable respuesta: la pastosidad (clasificación de los jueces).

Dentro de las variables de calidad se han separado como variables estáticas aquellas variables que eran pura información acerca del jamón, como el día de medición, puesto que son constantes a lo largo de todo el proceso.

En este trabajo se realizará un pretratamiento correspondiente a cada bloque de variable y se mostrará si es conveniente medir todas estas variables por su influencia en la pastosidad de los jamones. Se crearán modelos exploratorios y de predicción con el fin de sustituir estos análisis destructivos para crear un procedimiento de clasificación basado en la predicción.

## 1.2 Objetivos del trabajo

Los objetivos de este trabajo son:

- Identificar las variables que más inciden sobre la pastosidad y estudiar las relaciones entre ellas.
- Comparar los modelos PLS y PLS-DA con los dos bloques de variables (calidad y proceso) para estudiar tanto diferencias en el pretratamiento de datos como en los coeficientes de bondad de ajuste y predicción (tasas de acierto).
- Estudiar la conveniencia de aplicar técnicas de Minería de Datos (MD) para la clasificación de los jamones atendiendo a su pastosidad. Para ello, será necesario transformar la variable respuesta en una variable categórica.
- Comparar los resultados obtenidos con PLS con los proporcionados por las distintas técnicas de MD, evaluándolos a través de matrices de confusión, tasas de acierto y cálculo de errores.

- Obtener conclusiones acerca del equipo de ultrasonidos utilizado en la medición de las variables de proceso. Estudiar si existe relación de las variables de proceso con alguna variable influyente en la predicción de la pastosidad.

# MATERIALES Y MÉTODOS

# 2.MATERIALES Y MÉTODOS

## 2.1 Parte experimental

Los datos obtenidos para la elaboración de este estudio provienen de medidas experimentales reales. Son fruto de una investigación para una tesis doctoral en el departamento de Tecnología de los Alimentos de la Universidad Politécnica de Valencia.

Los datos que se obtuvieron en un primer momento fueron medidas que correspondían a diferentes bloques de variables que posteriormente se separaron. El proceso para la obtención de las variables de proceso fue a través de un equipo ultrasonidos y las variables que se midieron una vez elaborado el producto final en el laboratorio fueron fruto de análisis destructivos de las muestras.

Las variables se han identificado como: variables estáticas (variables que no cambian con el tiempo y únicamente proporcionan información sobre el tipo de jamón, se mantienen constantes durante todo el proceso), variables de proceso (variables procedentes del equipo de ultrasonidos) y variables de calidad (variables procedentes de análisis de laboratorio). Por último, la variable respuesta pastosidad, que presenta una dualidad entre variable categórica y numérica.

### 2.1.1 Variables estáticas (Z)

Estas variables son constantes a lo largo del tiempo, es decir, se mantienen fijas en todas las fases del proceso (*k*). Por tanto, corresponden a información sobre el jamón. Presentan una estructura de datos bidireccional. Posteriormente se indicarán cuáles son (figura 5)

### 2.1.2 Variables de proceso (X)

Para las mediciones de proceso se utilizó un sistema de ultrasonidos, un instrumento que a través de suave contacto con la muestra es capaz de conocer el espesor del jamón. Durante todas las fases del proceso se tomaron medidas de 197 jamones (individuos). El proceso se puede separar en las siguientes fases:


A continuación, se muestra el funcionamiento del equipo de medida utilizado en cada fase o instante del proceso (*k*).

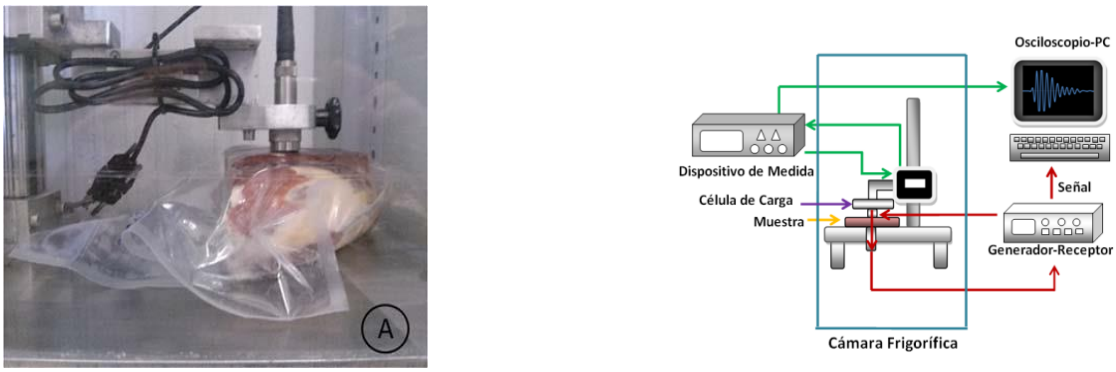
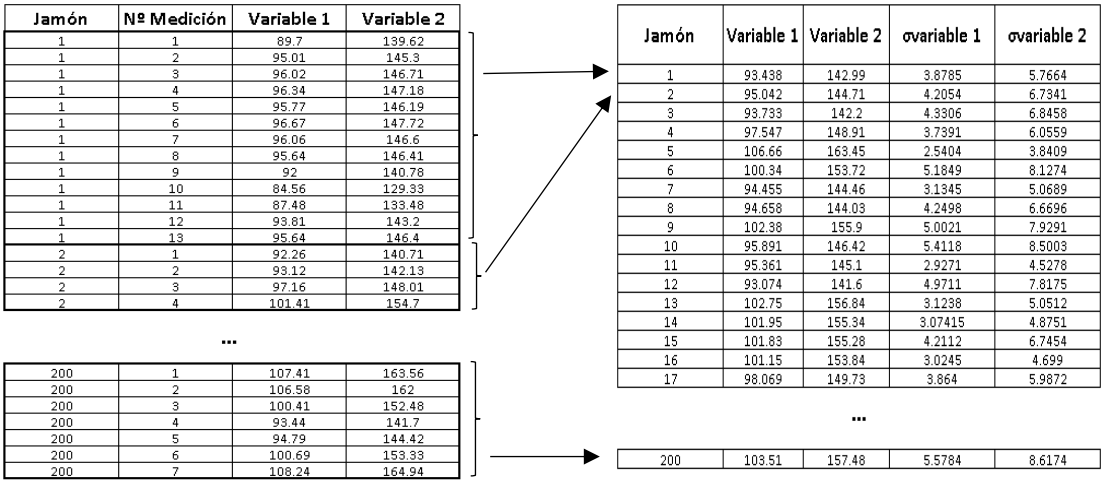



*Figura 1. Proceso de recepción de datos de proceso y equipo de medición* (Ruiz, 2015).

De izquierda a derecha de la figura 1, se muestra una imagen tomada de una muestra en el equipo de ultrasonidos, con su adaptador y la placa que medirá el espesor. Al lado, se muestra un diagrama del procedimiento de recepción de datos en cada medida. Comienza por medir el espesor y trasmitir la señal en un tiempo determinado (tiempo de vuelo) hasta llegar al receptor y que se registre el dato. El equipo ultrasonidos es capaz de conocer el espesor de la muestra a través del suave contacto de dos placas y emitir una señal sobre la medida. Las variables que se obtuvieron fueron: tiempo de vuelo (tiempo que tarda en recibir la medida), velocidad ultrasonidos (velocidad con la que la placa desciende hasta la muestra), espesor (cantidad de masa expresada en cm) y AV (incremento de velocidad, expresado como la diferencia de la velocidad ultrasonidos medida menos el promedio de la fase de frescos).

A través de un análisis exploratorio de los datos, se concluyó que los datos de proceso se iban a reorganizar sobre una estructura tridireccional, es decir, una estructura "Three-way". Para obtener esta estructura tridireccional, se empleó el programa "*Matlab 2018*" (Thompson & Shure, 1995) para reorganizar los datos. Las mediciones proporcionadas por el equipo que las realizó no aseguraban el mismo número de repeticiones para cada punto del jamón, ni que fueran tomadas exactamente siempre en el mismo punto, aunque sí de manera aproximada. El músculo elegido para la realización de las mediciones fue el Bífidus femoral, ya que es uno de los que más información puede aportar, según los expertos. Por esta razón, se realizaron medias de un mismo jamón, así como sus desviaciones típicas, a partir de las diferentes mediciones realizadas sobre cada jamón, con el fin de que todos los jamones (observaciones) tuvieran el mismo número de variables y mediciones. La Figura 2 presenta un esquema del proceso seguido para la creación de la estructura de datos:



| Jamón | Nº Medición | Variable 1 | Variable 2 |
|---|---|---|---|
| 1 | 1 | 89.7 | 139.62 |
| 1 | 2 | 95.01 | 145.3 |
| 1 | 3 | 96.02 | 146.71 |
| 1 | 4 | 96.34 | 147.18 |
| 1 | 5 | 95.77 | 146.19 |
| 1 | 6 | 96.67 | 147.72 |
| 1 | 7 | 96.06 | 146.6 |
| 1 | 8 | 95.64 | 146.41 |
| 1 | 9 | 92 | 140.78 |
| 1 | 10 | 84.56 | 129.33 |
| 1 | 11 | 87.48 | 133.48 |
| 1 | 12 | 93.81 | 143.2 |
| 1 | 13 | 95.64 | 146.4 |
| 2 | 1 | 92.26 | 140.71 |
| 2 | 2 | 93.12 | 142.13 |
| 2 | 3 | 97.16 | 148.01 |
| 2 | 4 | 101.41 | 154.7 |
| ... | | | |
| 200 | 1 | 107.41 | 163.56 |
| 200 | 2 | 106.58 | 162 |
| 200 | 3 | 100.41 | 152.48 |
| 200 | 4 | 93.44 | 141.7 |
| 200 | 5 | 94.79 | 144.42 |
| 200 | 6 | 100.69 | 153.33 |
| 200 | 7 | 108.24 | 164.94 |

| Jamón | Variable 1 | Variable 2 | ovariable 1 | ovariable 2 |
|---|---|---|---|---|
| 1 | 93.438 | 142.99 | 3.8785 | 5.7664 |
| 2 | 95.042 | 144.71 | 4.2054 | 6.7341 |
| 3 | 93.733 | 142.2 | 4.3306 | 6.8458 |
| 4 | 97.547 | 148.91 | 3.7391 | 6.0559 |
| 5 | 106.66 | 163.45 | 2.5404 | 3.8409 |
| 6 | 100.34 | 153.72 | 5.1849 | 8.1274 |
| 7 | 94.455 | 144.46 | 3.1345 | 5.0689 |
| 8 | 94.658 | 144.03 | 4.2498 | 6.6696 |
| 9 | 102.38 | 155.9 | 5.0021 | 7.9291 |
| 10 | 95.891 | 146.42 | 5.4118 | 8.5003 |
| 11 | 95.361 | 145.1 | 2.9271 | 4.5278 |
| 12 | 93.074 | 141.6 | 4.9711 | 7.8175 |
| 13 | 102.75 | 156.84 | 3.1238 | 5.0512 |
| 14 | 101.95 | 155.34 | 3.07415 | 4.8751 |
| 15 | 101.83 | 155.28 | 4.2112 | 6.7454 |
| 16 | 101.15 | 153.84 | 3.0245 | 4.699 |
| 17 | 98.069 | 149.73 | 3.864 | 5.9872 |
| ... | | | | |
| 200 | 103.51 | 157.48 | 5.5784 | 8.6174 |

*Figura 2. Estructura de los datos experimentales de proceso*

Este tipo de estructura de datos mostrada en la figura 2 se obtuvo para cada instante del proceso. Por tanto, hubo que realizar los cálculos para cada instante (fase del proceso de curación). Posteriormente, se organizaron los datos en forma de cubo y se realizó un desdoblamiento de tipo "Batch wise" con el fin de poder conocer la dinámica de las variables en todo el proceso. Tras haber transformado el cubo en una estructura bidireccional, nos permitió realizar los análisis basados en estructuras latentes así como los modelos empleando técnicas de aprendizaje supervisado.

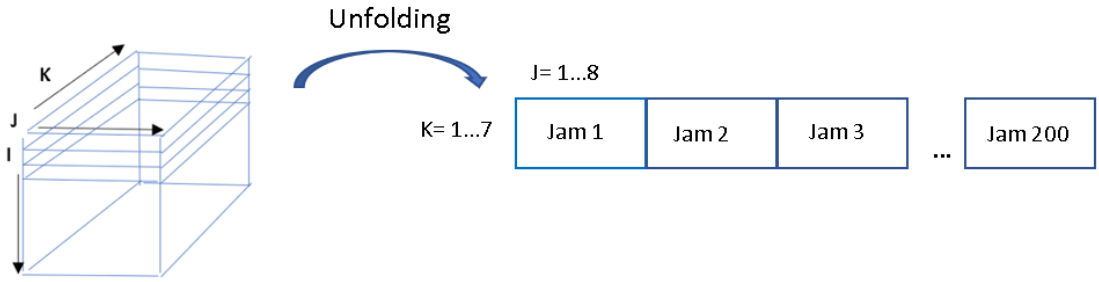



*Figura 3. Desdoblamiento Batch-Wise de la estructura Three way.*

Siendo:
I el número de observaciones que en este caso es el número de jamón [1,200].
J el número de variables de proceso, [1,8]
*K* el número de instantes de tiempo, en este caso fases del proceso. [1,7]

Aunque inicialmente se tuvieran 200 jamones, fueron eliminados 3 jamones por detección de errores experimentales antes de realizar los modelos. Los jamones descartados fueron el 14,33 y 127.

### 2.1.3 Variables de calidad (X)

Los datos de calidad son fruto de ensayos destructivos de las muestras. Las mediciones se realizaron a partir de lonchas del jamón, no del jamón entero como en las variables de proceso. Los parámetros corresponden a variables sensoriales, fisicoquímicas y bioquímicas. La estructura de datos es bidireccional, apta para realizar todos los análisis correspondientes. Se muestra a continuación una muestra de cada loncha obtenida para cada jamón.

*Figura 4. Muestra destructiva experimental de las mediciones de calidad*

En la figura 4 se pueden observar diferentes músculos del jamón así como los puntos donde realizaron las mediciones. La parte recuadrada corresponde al músculo Bífidus Femoral. Todas las mediciones fueron realizadas en este músculo y también se realizaron varias medidas obteniendo como dato final la media de dichas mediciones.

### 2.1.4 Variables respuesta (Y)

Se tenían 3 variables respuesta, relacionadas entre sí. La variable respuesta que más información aportó fue la pastosidad numérica. Esta variable corresponde a una media de 3 jueces expertos catadores en jamón. Las otras dos variables respuesta, provienen de un tipo de clasificación dependiente de dos variables de calidad: una medida de fuerza (Y90) y otra de variable sensorial. Se realizaron análisis exploratorios en función de la variable respuesta. La conclusión fue que la pastosidad numérica era la que más información aportaba con respecto a todas las variables explicativas y los siguientes modelos se realizaron a partir de esta

Con tal de esclarecer las posibles dudas sobre las diferentes variables y sus estructuras de datos, se muestran a continuación de forma resumida las siguientes figuras 5 y 6.

| ESTÁTICAS |
|---|
| Xz |
| Días salado |
| Peso inicial |
| Día Medida1 |
| Día Medida2 |
| Día Medida3 |
| Día Medida4 |
| Día Medida5 |

| PROCESO |
|---|
| Xproceso |
| Tiempo vuelo |
| Espesor |
| Velocidad |
| ΔV |
| Desv tiempov |
| Desv esp |
| Desv veloc |
| Desv ΔV |

| CALIDAD |
|---|
| Xcalidad |
| Merma |
| PH |
| F0 |
| Desv F0 |
| Y2 |
| Desv Y2 |
| Y90 |
| Desv Y90 |
| Adhesividad Sensorial |
| Desv Adh Sensorial |
| Viscosidad Saliva |
| Desv Viscos Saliva |
| Sal |
| Humedad |
| Adhesividad |
| Nitrogeno total |
| Nitrogeno No proteico |
| Indice Proteólisis |

| RESPUESTA |
|---|
| Y |
| Pastosidad numérica |
| Pastosidad (clase) |

*Figura 5. Variables constituyentes de cada uno de los bloques de variables.*

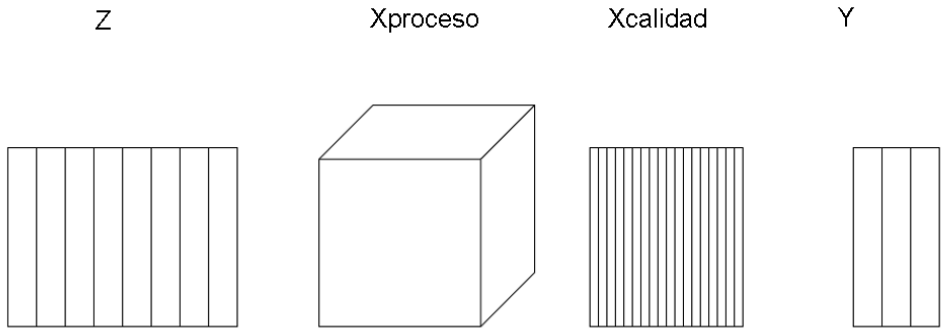



*Figura 6. Estructuras de datos*

## 2.2 Preprocesado de datos

La fase de preprocesado tiene mucha influencia en los posteriores análisis que se desarrollen. Es posible que determinados preprocesados poco comunes proporcionen mejores resultados que preprocesados más típicos, tales como un autoescalado a varianza unitaria.

Por esta razón, es muy importante utilizar una metodología previa antes de tratar los datos como conocer de que naturaleza son, realizar análisis exploratorios, es decir, conocer los datos

para posteriormente conocer el tipo de preprocesado que necesitan nuestros datos. Este es un punto primordial del análisis que podrá condicionar de forma significativa los resultados.

Este hecho es puesto de manifiesto por numerosos autores, sobre todo en el caso de matrices *3-way*, donde el autoescalado de las mismas destruye la estructura trilineal (Westerhuis, Kourti, & MacGregor, 1999). Consideran que el preprocesado de las variables debe llevarse a cabo según la problemática que se pretende resolver y no en función del sesgo que dicho preprocesamiento pueda introducir en los resultados proporcionados por el modelo que se esté utilizando. De hecho, concluyen que el preprocesado que se le establezca a los datos es más importante que los futuros modelos con carácter predictivo que se puedan dar, a pesar de que los modelos difieran en términos de interpretabilidad.

### 2.2.1 Autoescalado

Cuando se realizaran modelos predictivos con técnicas multivariantes o de aprendizaje supervisado que posteriormente se detallarán, es habitual centrar los datos para cada variable con el fin de que los valores sean comparativos. Se resta a cada observación la media de todas las observaciones para cada variable, de forma que la media de todas las variables es 0. De esta forma, centrar consiste simplemente en cambiar la escala de la variable, pero lo que no cambia es la distancia entre las puntuaciones de la variable.

El proceso de centrado establece el centro de coordenadas en el espacio de las variables originales en el centro de gravedad de la nube de puntos que generen los datos. Complementariamente al centrado, se suele escalar las variables a varianza unitaria, ya que las variables presentan diferentes tipos de variabilidad y de esta forma se unifica.

Además, tal y como ocurre habitualmente en datos recogidos de industrias, reales, las diferentes variables tienen diferentes unidades de medida. Por tanto, a veces es necesario realizar un autoescalado (centrado y varianza unitaria).

$$x_{ijk}^{centrado} = x_{ijk} - \bar{x}_{jk} \quad Ec.1$$

$$\bar{x}_{jk} = \frac{\sum_{i=1}^{I} x_{ijk}}{I} \quad Ec.2$$

El escalado de las variables asegura que las variables que más peso tienen no ejerzan mayor influencia sobre el modelo. Se estandarizan las variables dividiendo cada variable por su desviación típica. De esta forma, todas las variables tienen varianza unitaria, contribuyendo así de la misma manera al modelo.

$$y_{jk}^{escalada} = \frac{y_{jk}}{\sigma_y} \quad Ec.3$$

Si realizamos un escalado a varianza unitaria por variables (escalado clásico) al disponer de una estructura tridireccional, la variable JK se trata como una variable independiente. Es decir, que las relaciones trilineales entre variables pueden quedar distorsionadas, y se puede incorporar ruido en el modelo.

Por ello, el escalado por columnas (varianza unitaria) puede ser, en determinados casos, una manera incorrecta de escalar si nos encontramos ante una estructura multilineal.

### 2.2.2 Escalado por bloques de variables

Una de las alternativas al autoescalado en estructuras three-way es el esclado por bloques. En el escalado por bloques, se escala la evolución de cada variable (asumiendo que está en la segunda dimensión) a lo largo de la tercera dimensión (en este caso el tiempo). Estas segunda y tercera dimensiones son las mostradas en la figura 3.

A través de este escalado, no introducimos el posible ruido asociado al ruido base en el modelo y todas las variables pesan lo mismo sobre la variable respuesta. Dicho de otro modo, si escalamos dentro de cada variable, instantes de tiempo que no aporten nada (dicho de otro modo, variables de ruido como instantes de tiempo asociado a ruido base de un sistema de medida) acabarían teniendo la misma relevancia que otros que sí aportasen, empeorando así el modelo.

Desplegando la estructura three-way de la figura 3 a partir de un enfoque variable-wise (es decir, transformando el cubo en una matriz de dimensiones *IK*x*J*, si escalamos por bloques de variables). Cada columna *JK* se divide por la raíz cuadrada de su cuadrado medio, se muestra en las siguientes ecuaciones (4 y 6):

$$RMS_j = \sqrt{\frac{\sum_{i=1}^{I} x^2{}_{ijk}}{IK}} \quad Ec.4$$

$$x^*_{ijk} = \frac{x_{ijk}}{RMS_j} \quad Ec.5$$

Si la matriz de datos está centrada con respecto a cada variable (columnas), la expresión de la ecuación 4 es aproximadamente igual a la desviación típica muestral, quedando de la siguiente forma la expresión (6):

$$x^*_{ijk} = \frac{x_{ijk}}{\sigma_Y} \quad Ec.6$$

### 2.2.3 Balanceo de datos

Un aspecto deseable de cualquier estructura de datos es que haya el mismo número de individuos pertenecientes a cada clase. Sin embargo, cuando se trata de datos reales, experimentales, esto no suele ocurrir. Este hecho dificulta la labor de los clasificadores y la eficacia disminuye. Los modelos de clasificación tienden a clasificar hacia la clase mayoritaria puesto que al haber un mayor número de individuos pertenecientes a esa clase también habrá un mayor número de individuos en la parte de entrenamiento y en la parte de validación, con lo que dicha clase acabará siendo mejor clasificada, en detrimento del resto de clases.

Para solucionar este problema, se ha recurrido a balancear mediante el programa Rstudio mediante un remuestreo con remplazamiento, "Boostraping" (Rodrigo, 2017).

Bootstraping trabaja mediante el remuestreo *N* veces con reemplazo desde el conjunto de datos para formar nuevas tablas de datos (Bootstraps).

Este procedimiento consiste en escoger una muestra de la clase e indicarle el número de individuos que se necesitan. A partir de la muestra original, el código remuestrea añadiendo los individuos hasta completar el número indicado, en este caso 197 jamones. Este procedimiento se ha realizado para cada clase de jamones (figura 7).

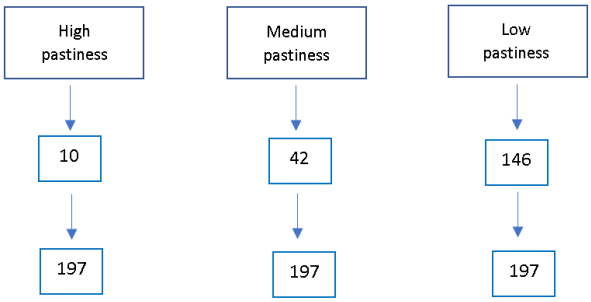



*Figura 7. Remuestreo de clases de los jamones*

Se han re muestreado hasta tener 197 jamones de cada clase para que el número de individuos estuviera en concordancia con los datos de proceso recogidos

## 2.3 Modelos utilizados

### 2.3.1 Modelos basados en estructuras latentes

#### *2.3.1.1. Análisis de Componentes Principales PCA*

Una de las herramientas básicas y principales basadas en estructuras latentes es el ***Análisis de Componentes Principales o Principal Component Analysis, PCA*** (Wold S. E., 1987)**.** Esta técnica soluciona problemas de los métodos multivariantes clásicos, principalmente los asociados a un

mal condicionamiento de la matriz de varianzas-covarianzas (problemas de invertibilidad de la matriz); permitiendo además trabajar con matrices de datos con un número mayor de variables que de individuos. El PCA puede ser utilizado para reducir la dimensión de variables perdiendo la menor cantidad de información, obteniendo así las nuevas variables latentes que son combinación lineal de las variables originales, e independientes entre sí.

El Análisis de Componentes Principales tiene como objetivo encontrar determinadas direcciones en el espacio de las variables que siguen las observaciones maximizando la varianza. Estas direcciones se les denominan componentes principales. La primera componente (dirección) será ortogonal a la segunda y así ocurrirá sucesivamente.

La interpretación de las componentes así como las relaciones entre las variables originales depende de la problemática a abordar. Es muy importante tener conocimientos previos al problema e investigar sobre el tema que se está abordando para tratar de obtener conclusiones coherentes y útiles.

Para emplear esta técnica es necesario disponer de matrices de datos bidireccionales. Sin embargo, cuando se tiene una matriz de datos "Three Way", es necesario desplegar la matriz antes de aplicar el PCA. En estos casos el análisis es conocido como "Unfold PCA" (U-PCA) (Camacho, Picó, & Ferrer, 2008). En la siguiente figura 8, se muestran los elementos que forman un PCA, común a al U-PCA, que únicamente se habría desdoblado para obtener estos elementos.

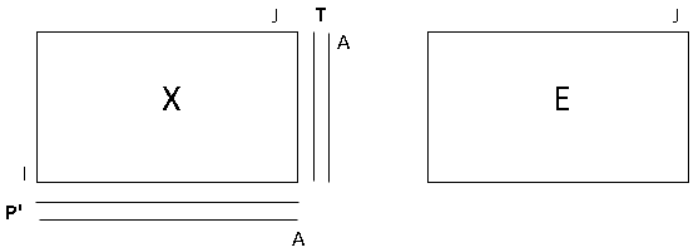



*Figura 8. Esquema ilustrativo de los elementos del PCA*

Siendo *I* el número de observaciones, *J* el número de variables y *A* el número de componentes principales. La matriz **T** es conocida como matriz de *scores*, resultado de las proyecciones de las observaciones sobre cada variable latente a partir de la matriz **P**, conocida como matriz de *loadings* o pesos, cuyas columnas son ortonormales y cuyos elementos reflejan el peso o relevancia de las variables originales sobre las componentes principales. La matriz **E** representa a la matriz que recoge para cada individuo la parte que no es explicada por cada una de las variables. No hay que menospreciar esta matriz de información, pues muchas veces la parte de los residuos nos puede aportar información sobre cómo está trabajando el modelo según los datos; ya que en ocasiones hay parte de la información de los datos que no se está viendo reflejada en las componentes.

### *2.3.1.2. Partial Least Squares*

Cuando además de tener variables explicativas disponemos de la variable respuesta en cuestión, es posible realizar modelos de predicción. Uno de ellos es la ***regresión de mínimos cuadrados parciales (Partial Least Squares Regression o simplemente Partial Least Squares, PLS).*** Esta técnica (Wold, Sjöström, & Eriksson, 2001) se utiliza en diversos ámbitos, y puede utilizarse de forma exploratoria para estudiar la posible relación de los datos cuando no hay un modelo causal conocido sobre la variable respuesta. Se crea un modelo de predicción a partir de las variables explicativas y se basa en la proyección de estructuras latentes.

El objetivo es extraer componentes (variables latentes) que son las direcciones sobre las que se proyectan las observaciones y que explican la máxima covarianza de la estructura interna de los datos con la de la/s variable/s respuesta. Por tanto, el objetivo es maximizar la covarianza con la variable respuesta (Ec. 7).

$$Cov\ (t, y) = r(t, y) \cdot S_t \cdot S_Y \quad Ec. 7$$

Siendo r(t,y) el coeficiente de correlación entre los scores (t) y la variable respuesta (y), $S_t$ y $S_y$ las desviaciones típicas de los scores y la variable respuesta.

La regresión PLS está compuesta por los siguientes elementos (figura 9):

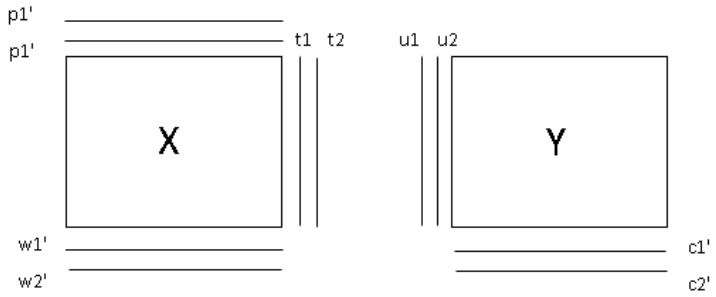



*Figura 9. Esquema ilustrativo de los elementos del PLS*

En este caso, se van a tener dos espacios: el de las **X** y el de las **Y**. Las componentes principales extraídas han de ser ortogonales en el espacio de las **X** e independientes entre sí, sin embargo, no tiene porque ocurrir en el espacio de las **Y**.

Las componentes PLS juntas forman hiperplanos en ambos espacios. Al proyectar las observaciones se obtiene los vectores *scores* para la primera ($\mathbf{t}_1$,$\mathbf{u}_1$) y a segunda componente ($\mathbf{t}_2$,$\mathbf{u}_2$) hasta A componentes ($\mathbf{t}_A$, $\mathbf{u}_A$).

Los *scores* (**t**) son las coordenadas definidas por el hiperplano del espacio de las **X**. Los *scores* **t** son un resumen de las variables **X** que están correlacionadas con **Y**. Los *scores* (**u**) son las coordenadas proyectadas en el espacio **Y**.

Los pesos, *loadings*(**p**) corresponden a las direcciones en el espacio de las **X**. Existen dos tipos de pesos: w y w*. Los *weights* (**w**) representan la correlación entre las variables **X** y las **Y**. En la 1ª dimensión, $\mathbf{w_1^*=w_1}$. En las siguientes dimensiones, los w* son los pesos que combinarán las variables originales **X** (no sus residuos, como con **w**) para formar los *scores* **t**, aplicando $\mathbf{W^*=W(P^TW)^{-1}}$. Las relaciones internas (**c**) representan los pesos que combinan variables **Y** para formar los scores **u** de forma que se maximice la correlación con las variables **X**.

#### *2.3.1.3. PLS-DA*

PLS requiere que la variable respuesta sea una variable numérica. Cuando la variable respuesta es categórica, como una clase o tipo, se utiliza la versión discriminante del PLS, conocida como PLS-DA. Esta variante del PLS utiliza el algoritmo de la regresión PLS para explicar y predecir la pertenencia de observaciones a varias clases, mediante la creación de una matriz **Y** que tiene tantas columnas como clases la base de datos. En cada columna, se asignan "1"s a los individuos asociados a la clase ligada a dicha columna, y "0"s al resto de individuos. Después, se construye el modelo como un PLS "normal".

El análisis discriminante PLS presenta muchas ventajas en casos donde no se puede aplicar en análisis discriminante clásico ya que, gracias a las características propias del PLS, permite utilizarlo cuando existe multicolinealidad entre las variables explicativas, cuando hay datos faltantes o cuando existe un mayor número de variables que observaciones. La estructura y el procedimiento es igual que la regresión PLS, la diferencia es que se crea un clasificador sobre la variable respuesta. Se muestra la estructura de los elementos que forman un PLS-DA en la siguiente figura 10.

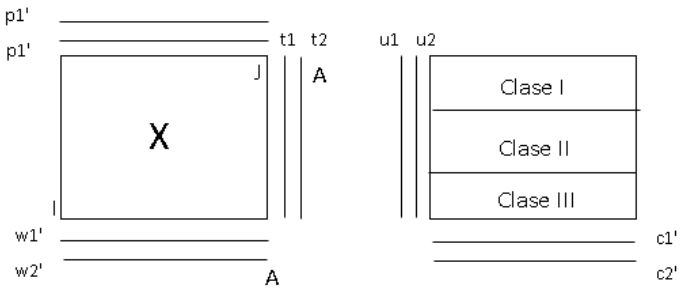



*Figura 10. Esquema ilustrativo de los elementos del PLS-DA*

### 2.3.2 Técnicas de aprendizaje supervisado

Tras utilizar técnicas basadas en estructuras latentes, se pretendía utilizar otro tipo de clasificadores para comparar la efectividad de todas las técnicas en la clasificación de los jamones pastosos. Para ello se utilizaron técnicas de aprendizaje supervisado.

Uno de los usos más extendidos del aprendizaje supervisado consiste en hacer predicciones del futuro basadas en comportamientos o características que se han visto en los datos ya almacenados (el histórico de datos). El aprendizaje supervisado es un tipo de algoritmo de Machine Learning que emplea un conjunto de datos conocidos para realizar predicciones. El conjunto de datos incluye datos de entrada y valores de respuesta. A partir de él, el algoritmo de aprendizaje supervisado busca crear un modelo que pueda realizar predicciones acerca de los valores de respuesta para un nuevo conjunto de datos.

En este caso, se van a utilizar modelos de clasificación, es decir, un sistema que predice una categoría. La clasificación supervisada es una de las tareas que más frecuentemente llevadas a cabo por los denominados sistemas inteligentes. Por lo tanto, un gran número de paradigmas desarrollados a través de la estadística son capaces de realizar las tareas propias de la clasificación.

En el siguiente apartado se van a explicar las técnicas que se han aplicado en este trabajo basadas en minería de datos como: Random Forest, K vecinos próximos, Naive Bayes, Support Vector Machine y árboles de clasificación.

El procedimiento común de estas técnicas (al igual que PLS) se basa en dividir una base de datos en 2 partes: entrenamiento ("*Training*") y validación ("*Test*"). Con la parte de "*Training*" se construye un modelo basado en los datos del pasado. Para validar el modelo, se crean predicciones, en este caso se predicen clases de los individuos a partir del modelo entrenado para posteriormente comparar con los datos que ya estaban clasificados en la parte de "*Test*". Posteriormente, se suele utilizar algún tipo de métrica para determinar la eficacia de las predicciones como matrices de confusión, tasas de acierto, cálculo de errores, etc.

#### *2.3.2.1. Árboles de clasificación*

**Los árboles de clasificación** se basan en algoritmos para clasificar que utilizan particiones sucesivas. Esta técnica presenta numerosas características pues permite comprender fácilmente las decisiones que toma el modelo según la estructura de datos que se disponga.

Los árboles de clasificación se basan en un proceso de división secuencial que tiene como origen la variable dependiente formando grupos homogéneos definidos mediante combinaciones de variables independientes en las que se incluyen la totalidad de los casos recogidos en la muestra. Básicamente, a partir de los individuos clasificados en la parte de entrenamiento, se construye el clasificador. Los árboles de decisión están compuestos por: nodos, ramas y hojas. Los nodos corresponden a las variables de entrada, las ramas representan los posibles valores de entrada y las hojas los de salida. Es muy importante la poda del árbol pues si los datos contienen incoherencias en la construcción, el modelo de predicción estará sobre ajustado. El procedimiento puede resumirse en los siguientes pasos:

Se dispone de una base de datos de entrenamiento que incluye información los individuos que pertenecen a cada clase. A partir de esta se construye el criterio de clasificación. Un criterio de

clasificación se basa básicamente en determinar cuáles son las variables con mayor poder discriminante, es decir, aquellas variables que mejor explican o separan a los individuos en la separación de cada clase.

- Se comienza con un nodo inicial y se divide el conjunto de datos en las partes más homogéneas sobre las clases utilizando una de las variables. A partir de esta variable utiliza como criterio (punto de corte) y se separan los datos en los conjuntos necesarios.
- Se repite el proceso en cada nodo dividiendo los datos según el nuevo criterio de la variable escogida y así sucesivamente.
- El proceso termina cuando se hayan clasificado todas las observaciones en cada grupo

Existen reglas para "validar" si el árbol está siendo bien construido. Se conoce como la pureza de nodo, el registro de si un nodo solo contiene observaciones de una única clase. También, hay criterios que se marcan previamente de la construcción del árbol, la profundidad y el umbral de soporte. La profundidad del árbol marca una cota de parada del proceso cuando se alcanza. De esta forma el árbol deja de construir nodos y ramas. El umbral de soporte especifica un número de observaciones mínimas para cada nodo, es una forma de considerar fiable un nodo a partir de la clasificación de un número determinado de observaciones.

Existen dos formas de poda: por coste-complejidad y la pesimista. La primera trata de equilibrar la precisión y el tamaño del árbol. La complejidad está determinada por el número de hojas que posee el árbol (nodos terminales). La poda pesimista utiliza los casos clasificados incorrectamente y obtiene un error de sustitución, eliminando los subárboles que no mejoran significativamente la precisión del clasificador.

#### *2.3.2.2. Random Forest*

**Random Forest** (Hastie, Tibshirani, & Friedman, 2001) es una técnica basada en crear árboles de clasificación de forma aleatoria. Introduce una aleatoriedad en cada nodo "p" de todas las variables y de éstas selecciona la mejor para realizar la partición. El procedimiento es el siguiente:

- Selección de individuos al azar (usando muestreo con remplazo) para crear diferentes sets de datos
- Se crean árboles seleccionando variables al azar en cada nodo del árbol dejando crecer al árbol sin podar.
- Crea un árbol de decisión con cada set de datos, obteniendo diferentes árboles, ya que cada set contiene diferentes individuos y diferentes variables.
- Predice los nuevos datos usando el "voto mayoritario", donde clasificará como "positivo" si la mayoría de los arboles predicen la observación como positiva.

Las técnicas "Árbol de clasificación" y "Random Forest" están basadas en el algoritmo *CART* (Classification and Regression Trees). Se basa en una partición recursiva, en cada iteración se selecciona la variable predictiva.

#### *2.3.2.3. K-nearest Neighbor (Knn)*

**K-nearest Neighbor** (Mora-Florez, Morales-España, & Barrera-Cárdenas, 2008) es un método de clasificación no paramétrico, es decir, no hace suposiciones sobre la distribución estadística que siguen los datos. Esta técnica clasifica nuevas observaciones como la clase mayoritaria entre los *k* vecinos más cercanos que se tengan en la base de datos de entrenamiento. Clasifica la nueva observación con los datos *k* más próximos conocidos, y dependiendo del parecido entre los atributos (variables) se ubicará en la clase que más se acerque al valor de sus propios atributos.

Con respecto, al número de vecinos que escoger, si se escoge un *k* muy grande, se corre el riesgo de que hacer una clasificación que esté condicionada a la mayoría y no a la similitud de las observaciones. Si se escoge un *k* muy pequeño puede haber imprecisión en la clasificación. Para enfrentar este problema se plantearon diferentes variaciones del método: en cuanto a la forma de determinar el valor de *k*, por ejemplo 1-nn, que usa como instancia de comparación al primer vecino más próximo encontrado. El procedimiento es:

- Definición de una medida de distancia entre puntos, habitualmente la distancia Euclidiana
- Cálculo distancias del punto a clasificar (nueva observación) respecto a todos los puntos de la muestra
- Selección de los puntos muestrales más próximos al que queremos clasificar
- Cálculo de la proporción de los k puntos que pertenece a cada una de las poblaciones
- Clasificación de la observación en la población con mayor frecuencia de puntos en los *k* vecinos más cercanos

Una práctica habitual es tomar $k=\sqrt{ng}$ donde *ng* es un tamaño de grupo promedio. Otra posibilidad es probar con distintos valores de *k*, aplicárselo a los puntos de la muestra cuya clasificación es conocida y obtener el error de clasificación en función de k. Escoger aquel valor de *k* que conduzca al menor error observado.

#### *2.3.2.4. Naïve Bayes*

**Naïve Bayes** (Chaparro, Giraldo, & Rondón, 2013) es un clasificador que presenta mucha simplicidad y rapidez. Construye modelos que predicen la probabilidad de pertenecer a una clase u otra. Las probabilidades se basan en el Teorema de Bayes, conocido como teorema de la probabilidad condicionada. Este teorema se refiere al cálculo de la probabilidad condicional del evento A dado que ha ocurrido el evento B, su forma general es: Si $A_1$, $A_2$,…,$A_n$ son eventos exhaustivos y exclusivos tales que $P(A_i)> 0$, $\forall$ i = 1,2 ,..., n, sea B un evento cualquiera del que se conocen las probabilidades condicionales $P(B|A_i)$, la probabilidad $P(A_i|B)$ viene dada por la ecuación 8.

$$P\left(\frac{Ai}{B}\right) = \frac{P\left(\frac{B}{Ai}\right)P(Ai)}{P(B)} \quad Ec.8$$

El procedimiento para la clasificación de un nuevo individuo está dividido en dos partes:

Creación del modelo

- Cálculo de probabilidades a priori de cada clase, P (c), a partir de la siguiente expresión

$$P(c)P(X1,\dots,Xp|c) = P(c)\prod_{i=1}^{p} P(Xi|c)\ \ Ec.9$$

- Realizar un recuento para cada clase.
- Normalización de los valores entre 0 y 1.

Clasificación nueva observación

- Para cada nuevo individuo que se tenga se calcula la probabilidad de cada valor en pertenecer a una clase u otra.
- Se aplica la fórmula de Naïve Bayes (Ec.8)

La técnica "Laplace Smoothing", es una variante de NaiveBayes. Ambas operan igual, con la diferencia es que en Naive Bayes el término de Laplace es igual a 0 y en "Laplace Smoothing" se le da un valor determinado.

$$Si\ P(indiv|clase) = \frac{recuentos\ individuo\ en\ la\ clase}{recuento\ total\ individuos}\ \ Ec.10$$

Se introduce el término de Laplace *(V)* para posibles recuentos igual a 0, ya que podría ocasionar problemas para el cálculo de la probabilidad. Quedaría la siguiente ecuación:

$$P(indiv|clase) = \frac{recuentos\ individuo\ en\ la\ clase + 1}{recuento\ total\ individuos + V + 1}\ Ec.11$$

***2.3.2.5. máquinas de Soporte Vectorial (SVM)***

**Las máquinas de Soporte Vectorial** (Vargas, Conde, Paccapelo, & Zingaaretti, 2012) son un conjunto de algoritmos de aprendizaje supervisado que pueden ser implementados para problemas de clasificación o regresión.

Los datos de entrada (puntos) son almacenados como un vector. Dado un conjunto de puntos (espacio), en el que cada uno de ellos pertenece a una de las dos o más posibles categorías, el procedimiento es el siguiente:

- Se parte de un conjunto de entrenamiento etiquetadas en sus pertenecientes clases
- Representación de dichas muestras en el espacio para poder separar las diferentes clases. De esta forma, cuando aparezcan nuevas observaciones de la base de datos de

validación se sitúen en correspondencia con las observaciones del modelo y puedan ser clasificadas correctamente en función de su proximidad.

En el concepto "separación óptima" reside la principal característica de esta técnica, pues SVM busca el hiperplano que tenga la máxima distancia con los puntos que estén más cerca de él mismo. Por eso mismo, se les conoce a las SVM como clasificadores de margen máximo.

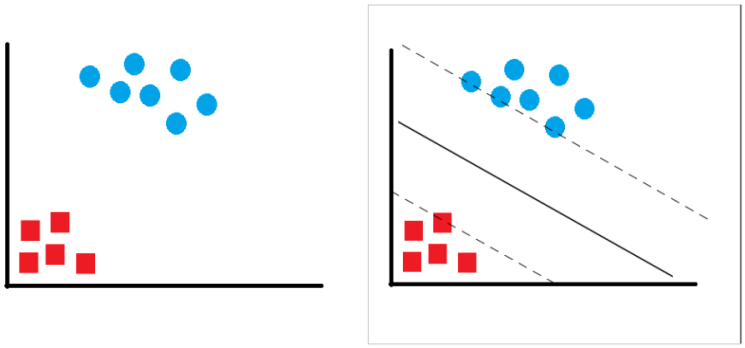


*Figura 11. Hiperplanos buscados para el caso de dos clases mediante la técnica SVM*

La manera más simple de realizar la separación es mediante una línea recta (figura 11), un plano recto o un hiperplano *N*-dimensional. Sin embargo, la realidad en los datos no suele ser tan sencilla, sino que un algoritmo SVM debe tratar con más de dos variables predictoras, curvas no lineales de separación, casos donde los conjuntos de datos no pueden ser completamente separados, clasificaciones en más de dos categorías. La representación por medio de funciones Kernel (función "ksvm") ofrece una solución a este problema, proyectando la información a un espacio de características de mayor dimensión el cual aumenta la capacidad computacional de la máquina de aprendizaje lineal permitiendo introducir diferentes tipos de funciones (lineales, gaussianas, polinomiales…).

### 2.3.3 Validación de modelos

Como resultado de aplicar todos los métodos de clasificación, existen varios criterios para validar la efectividad de los modelos. Uno de ellos es la tasa de acierto a partir de la matriz de confusión generada. Una matriz de confusión trata de explicar la cantidad de predicciones que se han clasificado como correctas e incorrectas. Existirán clases que se habrán clasificado como otras clases y por tanto el modelo no habrá acertado y clases que se han clasificado correctamente y contribuyen a que la tasa de acierto aumente. Se muestra de forma visual una matriz de clasificación. En este caso, se ha elegido una matriz de 3 tipos de clases porque

el problema abordado y los modelos empleados han tratado de clasificar 3 clases de pastosidad (Tabla 1).

*Tabla 1. Matriz de confusión de tres clases*

| real / pred | Y=0 | Y=1 | Y=2 |
|---|---|---|---|
| $\hat{Y}$=0 | P11 | P12 | P13 |
| $\hat{Y}$=1 | P21 | P22 | P23 |
| $\hat{Y}$=2 | P31 | P32 | P33 |

Donde P11, P22 y P33 corresponderán a predicciones correctas (clasificadas como 0, como 1 y como 2), mientras que P12, P13, P21, P23, P31 y P32 corresponderán a predicciones erróneas (predicciones que han clasificado a individuos en una clase que no correspondían realmente). A partir de estos valores, se puede definir la tasa de acierto como el cociente entre las predicciones correctas y el total de las predicciones realizadas. Es decir:

$$Tasa\ de\ aciertos = \frac{P11 + P22 + P33}{P11 + P22 + P33 + P12 + P13 + P21 + P23 + P31 + P32} \quad Ec. 12$$

$$Tasa\ de\ fallos = 1 - tasa\ de\ aciertos \quad Ec. 13$$

Una forma de validar los modelos es empleando la técnica "Hold Out iterativo", consiste en el siguiente procedimiento:

- Se parte la base de datos en entrenamiento (66, 67 % de los datos en este trabajo) y validación (33,33%) de forma aleatoria.
- Se crean los modelos de clasificación a partir del set de entrenamiento y las predicciones con los datos de validación.
- Se calcula una medida de bondad de ajuste, en este caso, la tasa de acierto.
- Se repite este procedimiento *N* repeticiones.
- Se calcula la media de las tasas de aciertos para cada técnica empleada y se evalúan resultados.

#### *2.3.3.1 Análisis de la Varianza (ANOVA)*

Para poder validar si hay diferencias estadísticamente significativas entre los métodos de clasificación utilizados para determinar la pastosidad de los jamones, se ha recurrido al Análisis de la Varianza (ANOVA) (Montgomery, 2005) es una técnica que estudia la significación estadística de uno o más factores al comparar las medias de los niveles de los factores. En

aquellos casos en los que el efecto de un factor resulta estadísticamente significativo, se han utilizado los intervalos de diferencias mínimas significativas o Least Significative Differences,

LSD, para ver entre qué niveles del factor (métodos en nuestro caso), existen diferencias (en este caso en el porcentaje medio de acierto en la clasificación).

## 2.4 Metodología empleada

La metodología empleada se puede resumir en el esquema de la figura 12. Esta metodología es la utilizada para los resultados mostrados en el siguiente capítulo.

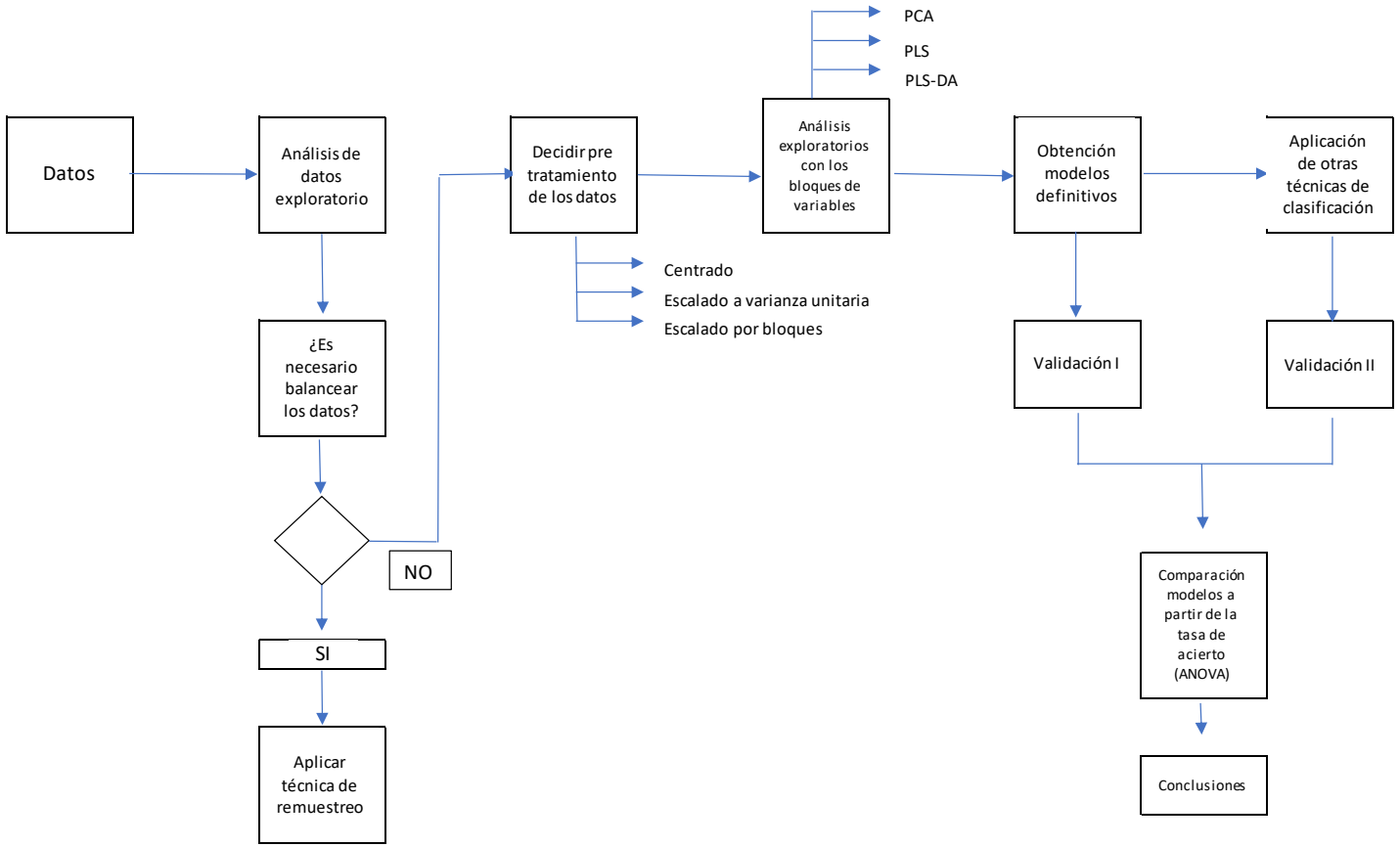



*Figura 12. Metodología empleada en el trabajo*

# RESULTADOS

# 3.RESULTADOS

## 3.1 PCA mediante desdoblado Batch Wise

El primer paso antes de realizar todos los análisis fue un PCA con las variables de proceso mediante un desdoblamiento de tipo "Batch wise" de los bloques. Este análisis nos permitió estudiar la dinámica de las variables de proceso a lo largo de cada instante de tiempo, así como las relaciones que había entre ellas a partir de sus trayectorias (figura 13).

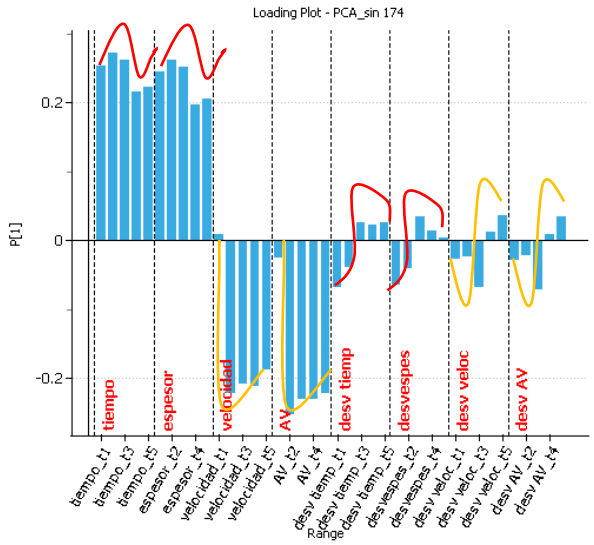



*Figura 13. Loadings de la primera componente*

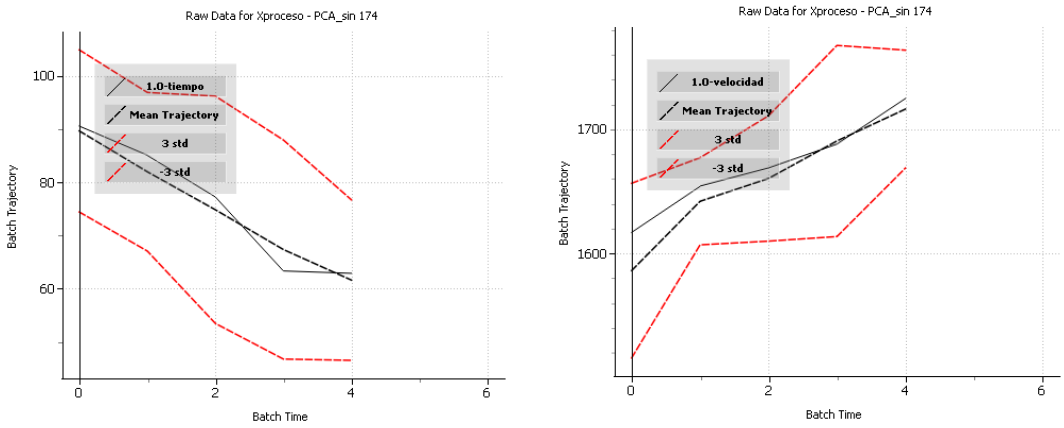



*Figura 14. Trayectorias del tiempo(derecha) y de la velocidad(izquierda)*

Las figuras 13 y 14 nos muestran la dinámica de las variables durante todas las fases del proceso, por lo que parece, los bloques tiempo-espesor y velocidad-AV se comportan de forma similiar, tienen una tendencia marcada. En la figura 13, en rojo está señalado las variables espesor y

tiempo, que mantienen una relación directa y en naranja las variables velocidad y AV que son inversas a el tiempo-espesor.

Gracias a este PCA exploratorio, se dedujo una relación indirecta entre dos bloques de variables: tiempo-espesor y velocidad-AV. Se visualiza claramente en las trayectorias.

## 3.2 Modelos PLS-1 y PLS-DA

La pastosidad numérica es una variable numérica que en función del rango en el que este puede decirse si un jamón es *High pastiness* [4,6], *Medium pastiness* [2,4[ y *Low pastiness* [0,2[. Esta variable se puede convertir en variable categórica puesto que está acotada para clasificar un jamón, por ejemplo, si un jamón presenta pastosidad numérica 2.4 pertenecerá a la misma clase que 3.1. Así mismo se ha transformado la variable pastosidad en variable categórica para realizar los siguientes modelos PLS-DA. De forma exploratoria, se han realizado modelos con todos los bloques de variables y diferenciado la forma de escalar con el fin de interpretar cómo varía la bondad de ajuste y predicción. Se muestran los modelos resumidos según la bondad de ajuste y predicción.

### 3.2.1 Modelos PLS

La siguiente tabla 2 muestra los resultados de bondad de ajuste de 3 PLS con los diferentes bloques de variables (proceso, calidad y estáticas) utilizando como variable respuesta la pastosidad numérica. El pretratamiento de datos se basa en un autoescalado (centrado y varianza unitaria) y sin balancear la base de datos, únicamente se han imputado los datos faltantes de las variables de calidad con el programa Rstudio. El objetivo ha sido probar PLS-1 independientes según los bloques de variables para poder obtener conclusiones hacia donde seguir la investigación. Hay que tener en cuenta que sólo las variables de proceso pueden estar influenciadas por el tipo de preprocesado (autoescalado o por bloques).

*Tabla 2. Resumen modelos PLS de la variable respuesta pastosidad numérica con autoescalado*

| Variable respuesta | Variables utilizadas | Núm. Comp | R2 (%) | Q2 (%) |
|---|---|---|---|---|
| Pastosidad numérica | Estáticas | 2 | 8.011 | 4.635 |
| | Proceso | 2 | 14.339 | 6.595 |
| | Calidad | 3 | 89.302 | 88.108 |

A partir de estos resultados mostrados en la tabla 2, podemos decir que las variables estáticas (pura información sobre el jamón) explican muy poco sobre la variabilidad de la variable respuesta. Las variables de calidad y de proceso, serán los bloques en los que se centrarán los siguientes análisis debido a los objetivos de este trabajo (se recuerda que se pretende evaluar la capacidad de las variables de proceso, no destructivas, para predecir la pastosidad de los jamones) y por los resultados obtenidos en estos modelos exploratorios. Además, se utilizará un

pretratamiento correspondiente acorde con los datos, así como un balanceo de datos previo puesto que existen grandes diferencias de individuos pertenecientes a cada clase.

***Análisis de las variables de proceso***

Los siguientes tres modelos PLS se han realizado a partir de una base de datos balanceada respecto del número de observaciones. Se han balanceado puesto que antes simplemente era un análisis exploratorio de los datos. Sin embargo, si se desea comparar los dos enfoques de PLS, PLS y PLS-DA, es necesario tener la misma base de datos, debido a las características propias de PLS. Con respecto al pretratamiento de los datos, para el análisis de los tres bloques de información, se han realizado dos modelos diferenciando la forma de escalar (autoescalado y escalado por bloques) para obtener conclusiones acerca de la forma en la que influyen los diferentes escalados. Con respecto a las variables de proceso que presentaban una estructura tridireccional, se ha seguido utilizando el desdoblamiento "Batch wise" para le estructura cúbica de los datos de proceso.

La bondad de ajuste mostrada en la tabla 3 muestra que los coeficientes R2 y Q2 de los modelos con los grupos de variables utilizados es alta, y tal y como se podrá comprobar más adelante, la bondad de ajuste empeora un poco si transformamos la variable respuesta en una variable categórica para realizar el PLS discriminante. Esto es evidente ya que se restringe el valor de la variable respuesta. Sin embargo, uno de los objetivos de este estudio se basa en estudiar la eficiencia del clasificador a partir de la tasa de acierto. Por otra parte, también se ha comprobado que escalando por bloques las variables de proceso, aumenta considerablemente la capacidad predictiva y explicativa del modelo.

En la siguiente tabla 3 muestra el objetivo de comparar cómo influye el tipo de escalado de las variables, así como el efecto de utilizar todos los bloques de información frente a sólo utilizar el bloque de proceso. En el caso de utilizar todos los bloques de información, se puede comprobar como si realizamos el PLS escalando a varianza unitaria la bondad de ajuste y predicción es casi la mitad que diferenciando la forma de escalar el bloque de variables de proceso.

Tal y como puede apreciarse en la tabla 3, aumentan los valores de bondad de ajuste según el preprocesado de datos que se realice. A través de estos resultados, podemos decir que las variables de proceso por sí solas no aportan mucha información ni valor de predicción para la pastosidad, independientemente del tipo de escalado que se haga. Sin embargo, las variables de calidad sí que predicen y explican muy bien la variable respuesta, tal como aparecía en la tabla 2.

*Tabla 3. Resumen modelos PLS de la variable respuesta pastosidad numérica*

| Variable respuesta | Tipo de escalado | Variables utilizadas | Núm. Comp | R2 (%) | Q2 (%) |
|---|---|---|---|---|---|
| Pastosidad numérica | Escalado por bloques | Proceso | 2 | 14.339 | 6.595 |
| | Autoescalado | Estáticas, proceso y calidad | 2 | 53.025 | 48.419 |
| | Autoescalado y escalado por bloques | Estáticas, proceso y calidad | 5 | 90.407 | 86.94 |

A partir de los análisis exploratorios anteriores, se decidió utilizar los bloques de proceso y de calidad únicamente, desechando las estáticas que no aportan capacidad predictiva, debido al objetivo ya comentado del trabajo. Sobre la forma de escalar, se separaron los modelos con respecto los bloques de variables. Se realizaron dos modelos PLS utilizando únicamente las variables de calidad (con autoescalado) y otro modelo utilizando además de las variables de calidad, las variables de proceso (escalando por bloques), se obtuvieron los siguientes resultados, equivalentes (tabla 4).

*Tabla 4. Resumen modelos PLS de la variable respuesta pastosidad numérica*

| Variable respuesta | Tipo de escalado | Variables utilizadas | Núm. Comp | R2 (%) | Q2 (%) |
|---|---|---|---|---|---|
| Pastosidad numérica | Autoescalado | Calidad | 3 | 92.51 | 92.07 |
| | Autoescalado y escalado por bloques | Calidad y Proceso | 5 | 93.474 | 92.565 |

De todas las pruebas realizadas, estos dos modelos son los que más información nos aportan para realizar las predicciones. Se presentan de forma individual los gráficos y resultados obtenidos

#### *3.2.1.1 Modelo con las variables de calidad*

En primer lugar, se introdujeron datos de la base de datos de calidad en *Rstudio* para que realizará una partición aleatoria de los datos para la construcción del modelo así como su posterior validación. Una vez obtenida la base de datos completa, se utilizó el programa *Aspen Pro MV* para realizar la regresión PLS así como sus posteriores predicciones.

A partir de los coeficientes de regresión, existían algunas variables que no eran estadísticamente significativas sobre la pastosidad en ninguna de las componentes extraídas. Así que, tras revisar los gráficos SPE y T2 de Hotelling para detectar observaciones atípicas y anómalas, y reconstruir los modelos, se eliminaron aquellas variables que no presentaban relación con la pastosidad. La detección de las variables no significativas se basó en descartar todas las variables cuyos

coeficientes de regresión obtenidos en un proceso de validación cruzada del modelo PLS no difieran significativamente de cero, es decir, presenten unos intervalos de confianza al nivel del 95 % que contengan el cero. Una vez eliminadas, se vuelve a realizar el modelo para comprobar si mejora la bondad de ajuste y predicción sin ellas en el modelo. En este caso, se eliminó la humedad y la adhesividad puesto que únicamente aportaban ruido en el modelo.

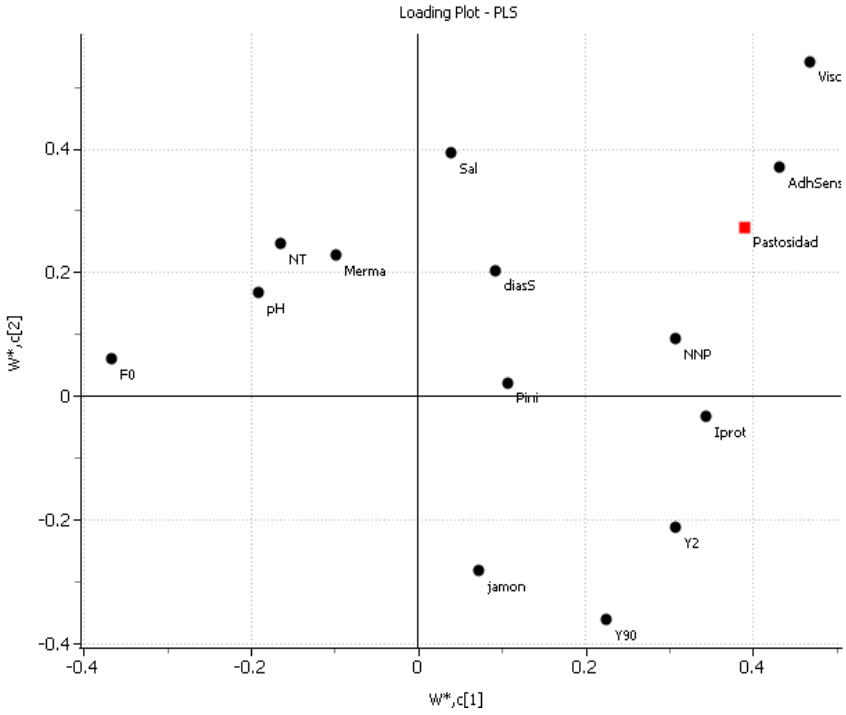



*Figura 15. Weights primera y segunda componente*

En la figura 15, se muestran los pesos de las variables sobre la pastosidad en las dos primeras componentes, que son las 2 que más explican y predicen. Podemos establecer relaciones directas que inciden sobre la pastosidad. El índice de proteólisis está muy relacionado con la primera componente, así como la viscosidad saliva o la adhesividad sensorial. Estas tres variables están próximas a la variable respuesta pastosidad; por tanto, mantienen una relación directa con ella. Así mismo, de forma indirecta el pH y el F0 (medida de dureza) influyen sobre la pastosidad. Cuánto más ácido sea el pH (pH<7) más efecto pastoso presentará un jamón.

Complementariamente, podemos volver a consultar los coeficientes de regresión (tras haber eliminado las variables no significativas comunes en las 4 componentes) para interpretar los efectos de cada parámetro sobre la variable respuesta (figura 16). Todas son estadísticamente significativas.

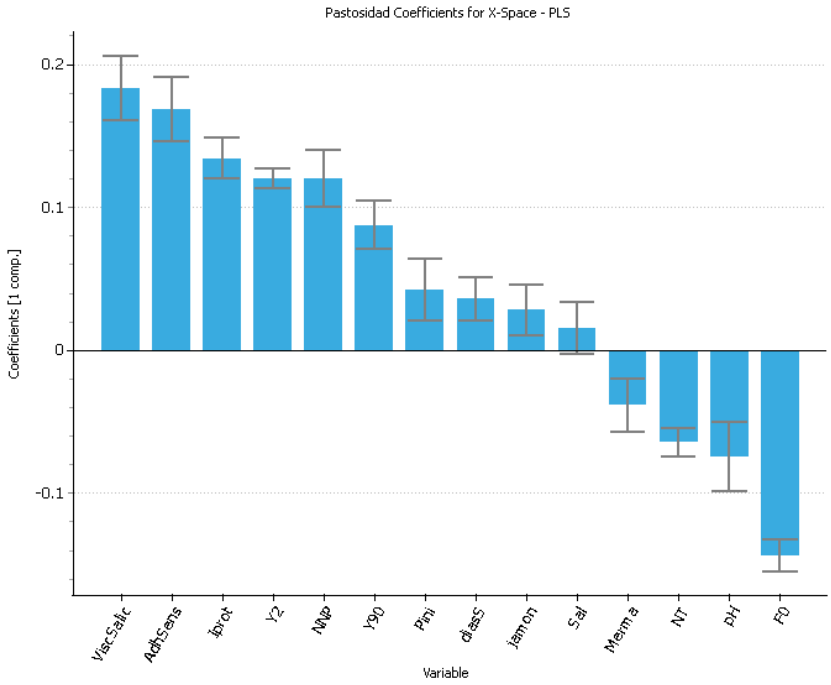



*Figura 16. Coeficientes de regresión del modelo PLS*

Se pueden establecer relaciones entre variables para futuras investigaciones de forma univariante, como es la relación que presenta el pH con la viscosidad saliva (variable influyente sobre la variable respuesta). Se han representado estas dos variables porque son las que más influyen sobre la pastosidad y presentan una relación que en un futuro se podría investigar. Se presenta en la figura 17.

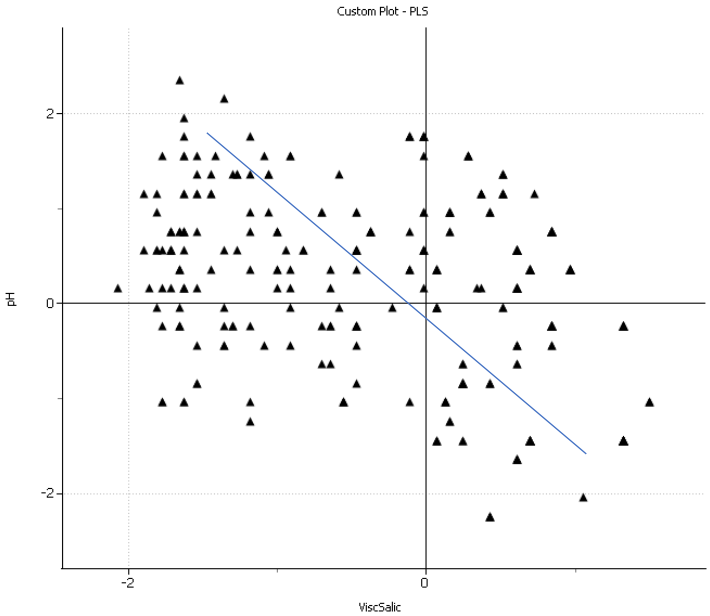



*Figura 17. PH vs viscosidad saliva*

A partir de los datos de validación y las predicciones realizadas por el modelo, se ha obtenido una matriz de confusión. Como la pastosidad numérica está acotada entre un rango de 0-6, dependiendo del valor estará clasificado en una u otra clase. Por tanto, con las predicciones numéricas que se han obtenido, se ha transformado según la clase en la que se encontrarían para poder obtener la matriz de confusión. Se muestran algunas de las predicciones así como la matriz de confusión calculada en las siguientes tablas 5 y 6.

*Tabla 5. Datos reales y predicciones del modelo PLS*

| N observación | Past real | Past pred | Clase real | Clase predicha |
|---|---|---|---|---|
| 395 | 0.25 | 0.19519985 | L | L |
| 396 | 2.625 | 2.54744035 | M | M |
| 397 | 2.625 | 2.54744035 | M | M |
| 398 | 2.625 | 2.54744035 | M | M |
| 399 | 0.125 | 0.07139772 | L | L |
| 400 | 0.16666667 | 0.11266509 | L | L |
| 401 | 0 | 0.05240441 | L | L |
| 402 | 0.5 | 0.44280411 | L | L |
| 403 | 0.125 | 0.07139772 | L | L |
| 404 | 0.375 | 0.31900198 | L | L |
| 405 | 0.66666667 | 0.60787362 | L | L |
| 406 | 0.625 | 0.56660624 | L | L |
| 407 | 6 | 5.8900979 | H | H |

*Tabla 6. Matriz de confusión obtenida para el PLS*

| real / pred | H | M | L |
|---|---|---|---|
| H | 34 | 9 | 0 |
| M | 23 | 52 | 8 |
| L | 0 | 5 | 66 |

En la tabla 6 se muestran las clasificaciones de los jamones que ha realizado el modelo PLS. La clase mejor clasificada ha resultado ser la *Low*, casi en su totalidad clasifica bien, tan sólo 8 individuos los clasifica como *Medium*. Teniendo en cuenta la poca variación que existe entre pertenecer a *medium* o *Los pastiness*, el error de predicción no es tan crítico que como si lo hubiera clasificado como *High*. Ocurre de la misma forma en los clasificados que son *High* como *Medium* pastiness, en este caso el modelo no comete ningún error al clasificarlos como la clase totalmente opuesta, *low*. La tasa de acierto es del 77,16%.

#### *3.2.1.2Modelo con variables de calidad y proceso*

Este modelo PLS se ha realizado a partir de las variables de calidad (autoescaladas) y las variables de proceso (escaladas por bloques). Como la base de datos ha aumentado por el número de

variables, se volvió a hacer un balance de datos, así como una partición aleatoria de los datos para posteriormente introducirlos en *Aspen*.

En este caso, no se eliminó ninguna variable de proceso aunque no resultara estadísticamente significativa. Como los datos de proceso procedían de una matriz desdoblada que reflejaba la evolución de las variables a lo largo de todo el proceso, se supuso que se perdía información y dinámica del proceso eliminando parte de los instantes de tiempo que no resultaban significativos. Por esta razón, evidentemente la bondad de ajuste y predicción es mayor que únicamente las variables de calidad ya que se introducen muchas más variables, aunque no todas ellas influyan en el modelo. El aumento es irrelevante como para concluir que estas variables predicen la pastosidad de un jamón.

Se presentan a continuación, algunos de los gráficos obtenidos para ilustrar el modelo.

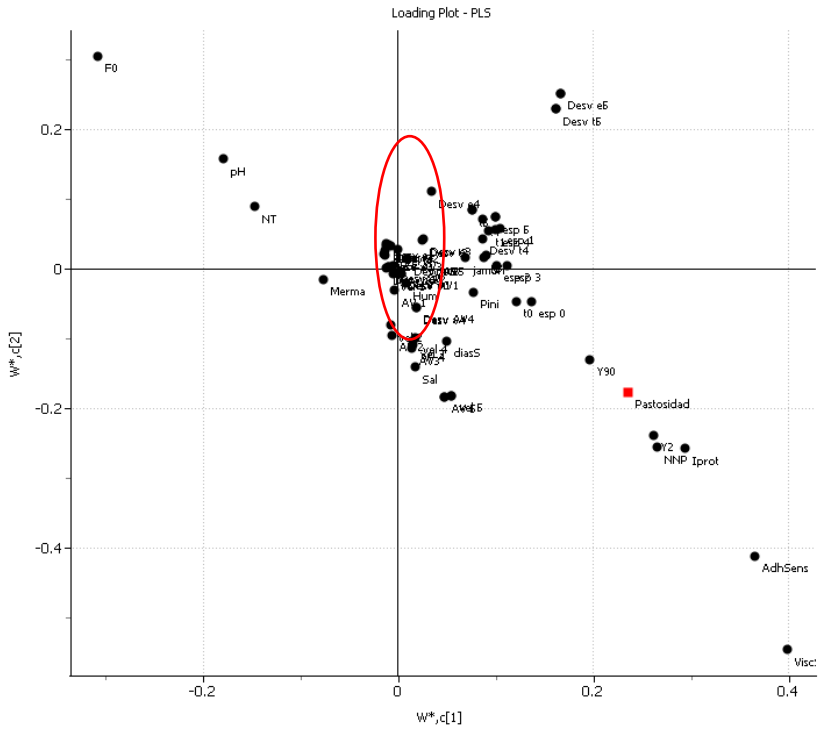



*Figura 18. Weights primera y segunda componente*

La figura 18, presenta los weights de las 4 componentes extraídas por el modelo. La zona marcada en rojo representa la poca significación que tienen las variables de proceso en el modelo. Se encuentran en el origen y agrupadas, prácticamente no explican la pastosidad de los jamones. Sin embargo, en la figura 18, además de mantenerse las conclusiones obtenidas sobre la influencia de las variables de calidad, aparecen algunas variables de proceso que parece que influyan de alguna forma sobre alguna variable de calidad, sin embargo, estas variables de calidad no son significativas sobre la pastosidad, por tanto, no se han encontrado relaciones directas con la variable respuesta.

De forma más visual, se adjunta el siguiente gráfico de la figura 19, donde se pueden apreciar las relaciones comentadas anteriormente.

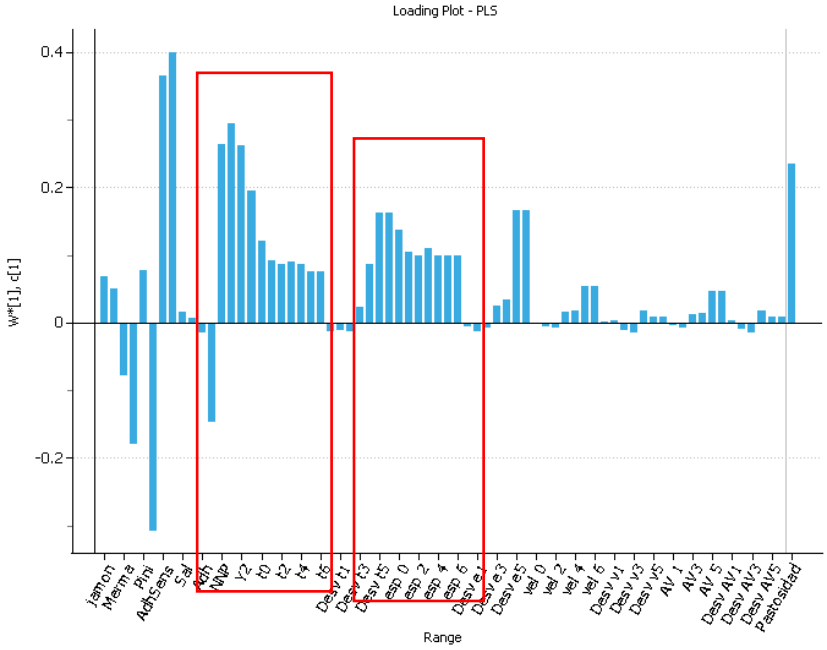



*Figura 19. Pesos de la primera componente del modelo*

En la figura 19, podemos observar cómo influyen las variables de proceso sobre la pastosidad. En concreto, los tiempos de vuelo del equipo ultrasonidos parecen tener una relación directa sobre la pastosidad del jamón y como habíamos comentado existe una relación directa entre el tiempo de vuelo y espesor, así como la velocidad y el AV. De hecho, parece que el tiempo de vuelo y espesor en el instante 1 (jamones frescos) incide de forma diferente sobre la variable respuesta sobre los demás instantes. Sin embargo, las variables de calidad siguen pesando más sobre la pastosidad y las variables de proceso no llegan a aportar suficiente capacidad predictiva por sí solas. De hecho, en la tabla 4 se observa un aumento muy bajo de la Q2 al introducirlas.

Con el fin de validar este modelo, se calcula la tasa de acierto a partir de la matriz de confusión obtenidas de las predicciones. Se adjuntan parte de las predicciones en la tabla 7, a nivel ilustrativo, y la clasificación realizada (tabla 8)

En este modelo, la clase peor clasificada es la *High*, que clasifica erróneamente como *Medium* 33jamones de los 65 que están clasificados como *High*. La tasa de acierto es del 75.13%. El resultado es mayor que únicamente con las variables de calidad, pero la diferencia es irrelevante. No se han encontrado relaciones directas sobre la pastosidad

*Tabla 7. Datos reales y predicciones del modelo PLS*

| N observación | Past real | Past pred | Clase real | Clase pred |
|---|---|---|---|---|
| 395 | 2.625 | 2.5758859 | M | M |
| 396 | 2.625 | 2.5758859 | M | M |
| 397 | 2.625 | 2.5758859 | M | M |
| 398 | 0 | 0.03749416 | L | L |
| 399 | 0.5 | 0.46029252 | L | L |
| 400 | 0 | 0.03749416 | L | L |
| 401 | 0.375 | 0.33584585 | L | L |
| 402 | 0.66666667 | 2.34422566 | L | M |
| 403 | 1.5 | 1.45586587 | L | L |
| 404 | 6 | 5.93594598 | H | H |
| 405 | 6 | 5.93594598 | H | H |
| 406 | 6 | 5.93594598 | H | H |
| 407 | 6 | 5.93594598 | H | H |

*Tabla 8. Matriz de confusión obtenida para el PLS*

| real / pred | H | M | L |
|---|---|---|---|
| H | 32 | 0 | 0 |
| M | 33 | 50 | 4 |
| L | 0 | 12 | 66 |

### 3.2.2 Modelos PLS-DA

Para la obtención de los siguientes modelos, se transformó la variable respuesta en categórica, es decir, en una clase. También se trató la base de datos balanceada así como su previa partición aleatoria para entrenamiento y validación. Se han analizado los modelos con los mismos bloques de variables que en los PLS-1: con las variables de calidad y con las de proceso y calidad. Se muestran a continuación los resultados en la tabla 9.

*Tabla 9. Bondad de ajuste y predicción para los modelos PLS discriminante*

| Variable respuesta | Tipo de escalado | Variables utilizadas | Núm. Comp | R2 (%) | Q2 (%) |
|---|---|---|---|---|---|
| Pastosidad (clase) | Autoescalado | Calidad | 4 | 44.189 | 41.063 |
| | Autoescalado y escalado por bloques | Calidad y proceso | 3 | 41.157 | 38.29 |

### *3.2.2.1Modelo con variables de calidad*

Tras introducir los datos, el programa extrajo 4 componentes para la regresión PLSDA. En la siguiente figura 20, se aprecia cómo contribuye cada variable según la componente. La primera componente, es la que más explica parte de las variables de calidad, engloba parte de las variables de carácter más sensorial (Viscosidad saliva y adhesividad), sin embargo, la segunda componente está más relacionada con variables químicas como la sal o la humedad.

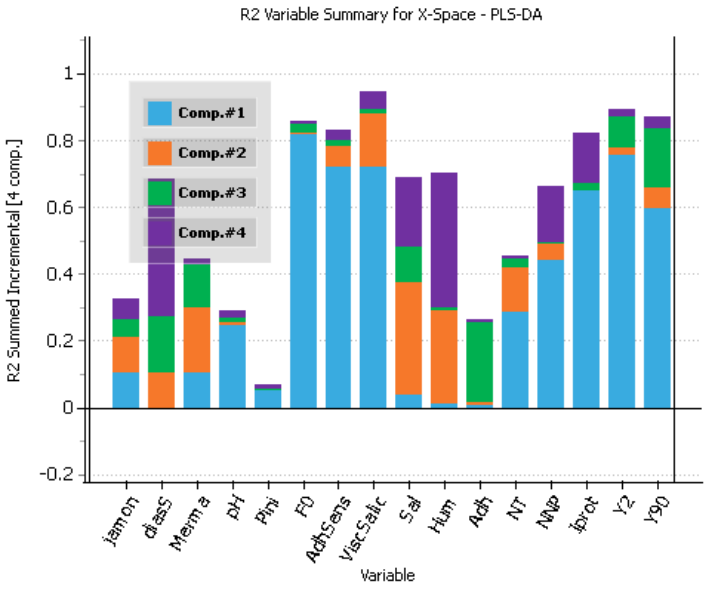



*Figura 20. Resumen pesos componentes según la variable*

Tras revisar los posibles datos anómalos que pudieran interferir en la formación de una nueva componente. Se comprueban a través de los coeficientes de regresión las variables que no son estadísticamente significativas basándonos en el criterio descrito en el apartado anterior. Tras la eliminación de algunas de ellas se muestran los resultados.

A partir de la figura 21, podemos decir que los centroides de las clases se discriminan de una manera más o menos clara. Las variables con poder discriminante son las mismas comentadas que en el modelo PLS con pastosidad numérica de las variables de calidad: siguen influyendo la viscosidad, el índice de proteólisis y la adhesividad sensorial en la clasificación de la pastosidad.

Las clases se discriminan de forma bastante clara, hay una clara diferencia entre la clase *Medium-High* y *Low*. Las conclusiones obtenidas sobre el efecto de las variables sobre la pastosidad son prácticamente iguales que en el PLS-1 de la pastosidad numérica. Las relaciones se mantienen, aunque la bondad de ajuste sea mucho menor que utilizando una variable respuesta numérica.

Se calcularon las predicciones que proporcionaba este modelo para mostrar la matriz de confusión siguiente, tabla 10.

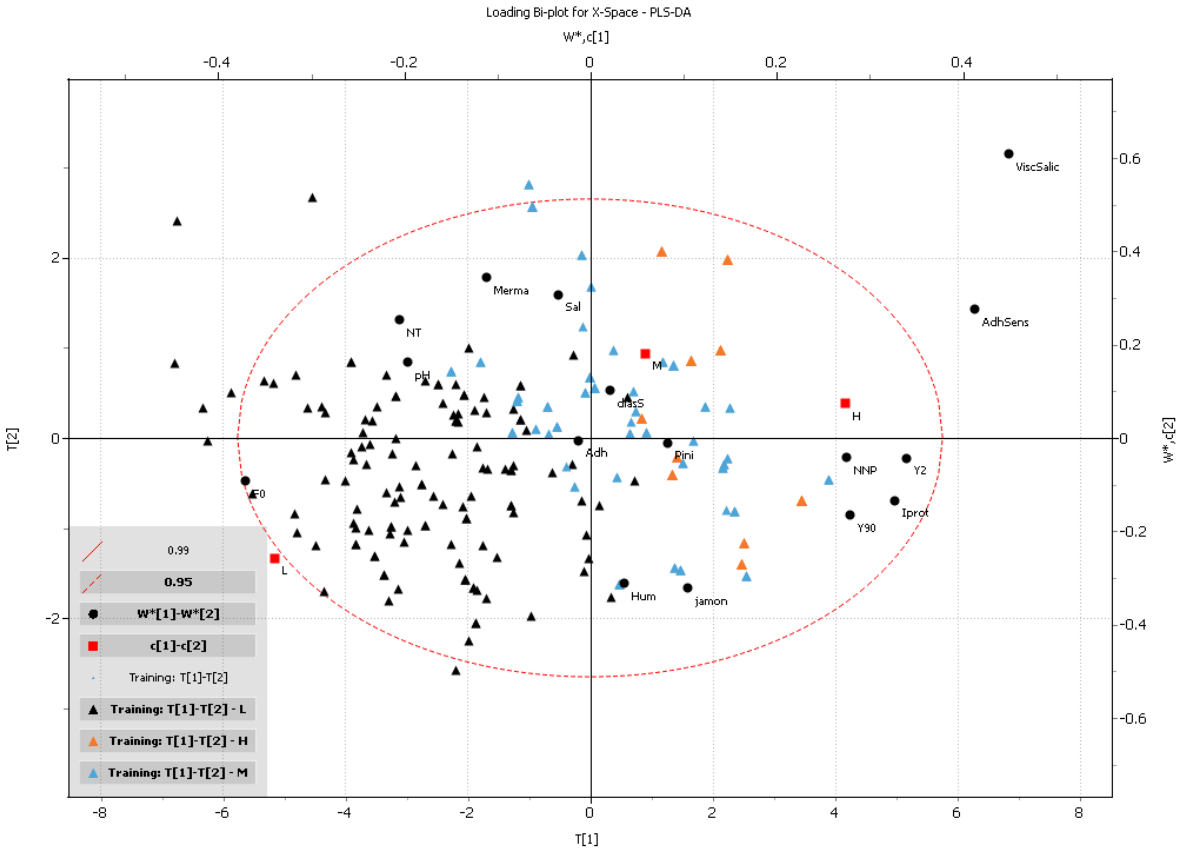



*Figura 21. Gráfico Bi plot del modelo PLSDA de las dos primeras componentes*

*Tabla 10. Matriz de confusión obtenida para el PLS-DA calidad*

| real / pred | H | M | L |
|---|---|---|---|
| H | 60 | 29 | 0 |
| M | 6 | 14 | 1 |
| L | 0 | 22 | 69 |

Este modelo clasifica muy bien la clase High, sin embargo, la clase peor clasificada es la Medium. Aún asi, su tasa de acierto (realizando el modelo únicamente una vez) es del 71.14%. Difiere muy poco del resultado obtenido con las mismas variables en el PLS.

### *3.2.2.2Modelo con variables de calidad y proceso*

Se muestran los resultados del PLS discriminante utilizando las variables de calidad y las de proceso.

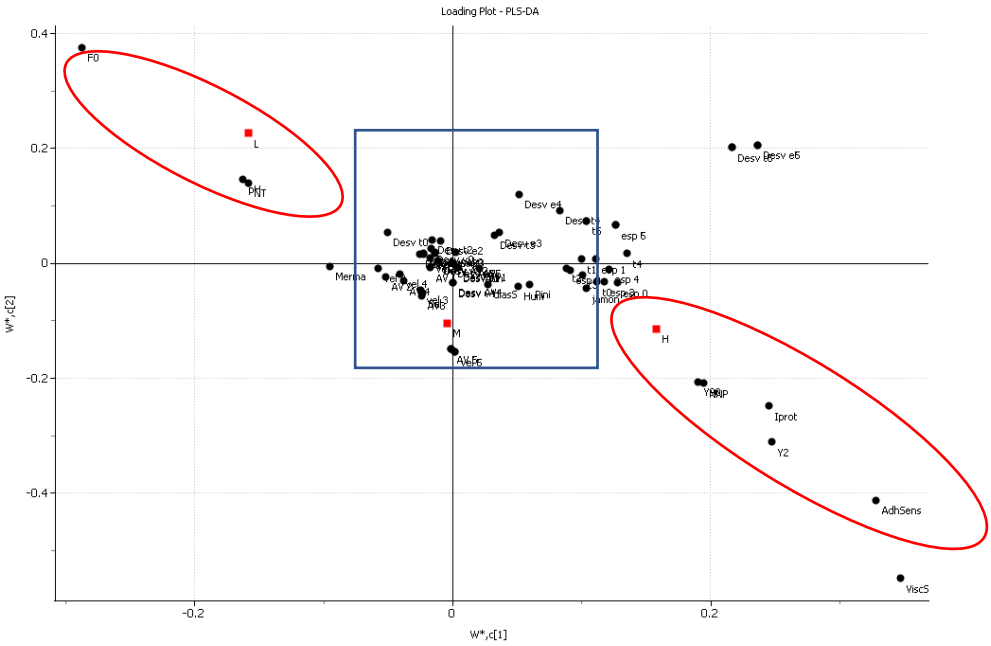



*Figura 22. Weightings primera y segunda componente*

Con respecto a esta figura 22 sobre los pesos de las dos primeras componentes, podemos decir que se mantienen las conclusiones sobre las variables que son significativas a la hora de clasificar o diferenciar las clases. Los dos círculos marcados sobre la figura reflejan las variables de calidad que discriminan esa clase. Por ejemplo, la F0 que es una variable de fuerza, discrimina la clase de baja pastosidad *Low*, sin embargo, las variables que se han comentado antes como por ejemplo Viscosidad o Índice de proteólisis son las discriminan la clase *High*. La clase *Medium* se relaciona con valores medios de estas variables.

Las conclusiones obtenidas sobre cómo influyen las variables de proceso sobre la variable respuesta son las mismas que en el modelo PLS-1 de las variables de calidad y proceso. Existen muchas variables en el centro del gráfico, es decir, variables que no son significativas y no influyen sobre la pastosidad.

También, se han calculado las predicciones y su respectiva matriz de confusión comparando la clasificación real con la predicha por el modelo. La tasa de acierto ha sido del 74.62%.

*Tabla 11. Matriz de confusión del modelo PLS-DA variables de calidad y proceso*

| real / pred | H | M | L |
|---|---|---|---|
| H | 57 | 28 | 1 |
| M | 9 | 18 | 2 |
| L | 0 | 10 | 72 |

Como podemos apreciar, la clase medium es la que peor se clasifica, tan solo 18 correctamente clasificados de los 56 jamones que están catalogados realmente como Medium. Este hecho tiene relación con la explicación que hemos dado en la anterior figura 22, es la clase que peor se

clasifica puesto que no tiene variables que la discriminen bien. Esta clase se encuentra en medio de los dos límites de clasificación.

## 3.3 Modelos basados en técnicas de aprendizaje supervisado

Con el fin de utilizar más técnicas en la clasificación de los jamones se han empleado técnicas de aprendizaje supervisado de minería de datos. Se ha utilizado la misma métrica de evaluación de la clasificación de los modelos, la tasa de acierto. El programa utilizado ha sido el Rstudio. Se han escogido los dos grupos de variables empleados en los análisis exploratorios PLS y PLS-DA.

### 3.3.1Modelos de predicción basados en las variables de calidad

Se ha utilizado la base de datos balanceada y sin datos faltantes que se ha elaborado previamente para todos los estudios con el fin de utilizar los mismos datos y que sean comparables. Por tanto, todos los siguientes modelos de clasificación se han realizado a partir 197 jamones para cada clase de pastosidad: *High* (H), *Medium* (M) y *Low* (L). Un total de 591 jamones.

Tras realizar los modelos exploratorios PLS y PLS-DA, se pretenden realizar modelos de predicción a partir de técnicas de minería de datos utilizando la variable respuesta pastosidad (transformada en variable categórica) creando así múltiples clasificadores mediante Rstudio.

Las técnicas utilizadas corresponden a técnicas de aprendizaje supervisado, es decir, técnicas que crean modelos a partir de patrones o comportamientos basados en una base de entrenamiento que después se validan con otra base de datos de validación. La técnica de validación utilizada ha sido el “Holdout repetido”, se realizan todos los modelos y se calcula la

tasa de acierto *N* repeticiones, en este caso 100 veces. En cada iteración se divide la base de datos para entrenamiento y validación. Se muestran a continuación, las matrices de confusión de la última iteración (100). Para cada iteración se han aplicado todas las técnicas, con el fin de poder utilizar posteriormente las mismas como un factor de bloqueo.

Con respecto al código utilizado, se adjunta en ANEXOS el código para los modelos creados a partir de las variables de calidad (“Aprendizaje supervisado calidad”, Anexo 6.1) y para las variables de calidad y proceso (“Aprendizaje supervisado calidad y proceso”, Anexo 6.2).

*Classification tree*

Las librerías utilizadas para crear esté árbol de clasificación han sido: Rpart (Thernau & Atikson, 2018), DMwR (Torgo, 2010), (Wickham & Bryan, 2018) , readxl (Wickham & Bryan, 2018) y xtable (Dahl, 2016). En la figura 23 se adjunta un árbol creado en una de las 100 iteraciones para ilustrar el procedimiento que sigue esta técnica. En primer lugar, a partir de los datos de entrenamiento se ha creado un criterio a partir de la variable que considera que más efecto tiene sobre la

variable respuesta pastosidad. Una vez creado este criterio y punto de corte comienzan una serie de órdenes lógicas para clasificar a los jamones.

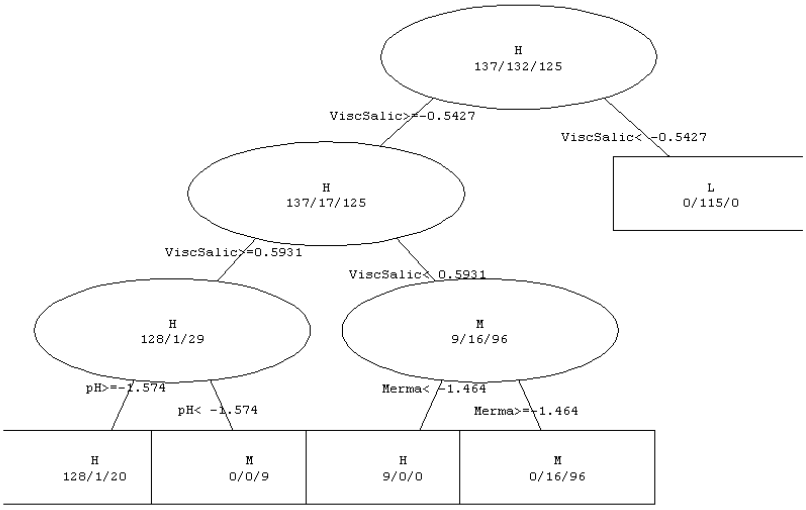



*Figura 23. Ejemplo de uno de los árboles de clasificación creados por el modelo*

*Tabla 12. Matriz de confusión iteración 100 Classification tree*

| real / pred | H | M | L |
|---|---|---|---|
| H | 64 | 5 | 0 |
| M | 0 | 71 | 5 |
| L | 0 | 0 | 52 |

Uno de los problemas que existen en los análisis que realizan los expertos (jueces) para clasificar si un jamón es pastoso o no, es que los umbrales de clasificación están tan cercanos y el criterio

es tan subjetivo, que puede crear conflictos de clasificación. Sin embargo, en la tabla 12 se muestra como este método (no destructivo) de predicción, ha clasificado en una iteración los 64 jamones *High* pastiness cuando realmente lo eran. Clasifica con muy baja tasa de error y con prácticamente cero una de las clases más críticas de la pastosidad.

*Random Forest*

La librería utilizada para esta técnica es: randomForest (Liaw & Wiener, 2002). (Venables & Ripley, 2002)Se ha utilizado una de las funciones de la librería para crear un clasificador a partir de los datos de entrenamiento. Posteriormente, se han calculado las predicciones utilizando el modelo creado y validándolo con los datos de test de la partición creada. Se muestra la tabla 13 de confusión de la última iteración.

*Tabla 13. Matriz de confusión iteración 100 Random Forest*

| real / pred | H | M | L |
|---|---|---|---|
| H | 64 | 0 | 0 |
| M | 0 | 76 | 4 |
| L | 0 | 0 | 53 |

Esta técnica está basada en árboles de clasificación, y prácticamente ofrece resultados muy parecidos a la técnica anterior. Clasifica todos los jamones *High* y *Medium* pastiness correctamente. Es importante que se clasifiquen los jamones como High si lo son, pero cuando son *Medium*, al ser una clase intermedia entre las clases opuestas es más difícil la clasificación por los expertos, sin embargo, "Random Forest" los clasifica prácticamente todos bien.

*Nearest Kneighbour*

Para emplear esta técnica se ha utilizado la librería "class" (Venables & Ripley, 2002). (Meyer, Dimitriadou, Hornik, Weingessel, & Leisch, 2017). Esta técnica crea predicciones a partir de los datos cercanos a la nueva observación proyectados en el espacio. En la creación del clasificador es necesario indicarle el número de datos muestrales (vecinos) que debe construir cuando se tenga una nueva observación, en este caso, hemos utilizado el siguiente criterio: $k = = \sqrt{N}$ , siendo *N* el número de datos pertenecientes a una clase. Como se dispone de una base de datos balanceada de 197 jamones para cada clase, los vecinos a consultar por el modelo (*k*) es igual a 14.

*Tabla 14. Matriz de confusión iteración 100 Vecinos más cercanos*

| real / pred | H | M | L |
|---|---|---|---|
| H | 64 | 32 | 2 |
| M | 0 | 39 | 5 |
| L | 0 | 5 | 50 |

En este caso, la tabla 14 refleja como la técnica vecinos más cercanos clasifica todos lo jamones High pastiness como la clase que son. Las otras dos clases están peor clasificadas, este hecho puede ser ya que cuando aparece una nueva observación el modelo se fija en los 14 vecinos más cercanos que tenga clasificando en la clase mayoritaria, al ser un número considerable de vecinos, puede que el error provenga de este hecho.

*Naive Bayes*

En este caso, el clasificador basado en el teorema de Bayes se ha construido utilizando la librería (Karatzoglou, Smola, Hornik, & Zeileis, 2004)"e1071" de Rstudio (Meyer, Dimitriadou, Hornik, Weingessel, & Leisch, 2017).

*Tabla 15. Matriz de confusión iteración 100 Naive Bayes*

| real / pred | H | M | L |
|---|---|---|---|
| H | 60 | 20 | 0 |
| M | 4 | 54 | 8 |
| L | 0 | 2 | 49 |

*Laplace Smoothing*

La técnica Laplace Smoothing es una variante de la técnica Naive Bayes, simplemente se modifica el valor de control de alisado de Laplace, un argumento de la función "naiveBayes". En Naive Bayes es el valor del argumento predeterminado (0), sin embargo, para esta técnica hemos utilizado Laplace=3 para comprobar si existen diferencias significativas al utilizar el control de alisado de Laplace.

*Tabla 16. Matriz de confusión iteración 100 Laplace Smoothing*

| real / pred | H | M | L |
|---|---|---|---|
| H | 60 | 20 | 0 |
| M | 4 | 54 | 8 |
| L | 0 | 2 | 49 |

En las tablas 15 y 16, se aprecia la misma clasificación. En esta iteración no existen diferencias entre ambas técnicas, sin embargo, cuando se presenten la tasan de acierto y se comparen las técnicas realizando todas las iteraciones podremos decir si existen diferencias estadísticamente significativas respecto a la media en la clasificación de la pastosidad.

*Support Vector Machine*

La técnica máquinas de soporte vectorial, crea predicciones a partir de la distancia de la creación de múltiples hiperplanos con respecto a los datos obtenidos. La librería utilizada en este caso es "kernlab" (Karatzoglou, Smola, Hornik, & Zeileis, 2004) (F, B, A, & K-A, 2017) (Mevik, Wehrens, & Hovde Liland, 2016), es un paquete extensible para métodos de aprendizaje basados en Kernel en R.

*Tabla 17. Matriz de confusión iteración 100 SVM*

| real / pred | H | M | L |
|---|---|---|---|
| H | 64 | 6 | 0 |
| M | 0 | 69 | 6 |
| L | 0 | 1 | 51 |

En este caso, para la última iteración del cálculo de las predicciones (tabla 17), la matriz de confusión muestra como clasifica el modelo creado por Support Vector Machine. Clasifica el 100

por cien de los jamones *High* pastiness correctamente. La clase peor clasificada es la clase intermedia *Medium* pastiness.

PLSDA

Para crear el PLS discriminante en Rstudio y poderlo introducir en el método de validación escogido (Hold Out repetido) se probaron diferentes librerías con el fin de comparar los resultados. El objetivo fue asegurarse de cómo realizaba las predicciones R. Se utilizaron las librerías: "mixOmics" (F, B, A, & K-A, 2017) , "pls" (Mevik, Wehrens, & Hovde Liland, 2016) y "DiscrimMiner" (Sanchez, 2013). Finalmente, se comprobó que los resultados fueron muy parecidos y se escogió la función "plsda" de la librería "mixOmics". El procedimiento fue similar a los anteriores, indicando la base de datos de entrenamiento, la variable categórica y creando las predicciones para su posterior comparación (tabla 18).

*Tabla 18. Matriz de confusión iteración 100 PLDA*

| real / pred | H | L | M |
|---|---|---|---|
| H | 57 | 0 | 37 |
| L | 0 | 55 | 16 |
| M | 7 | 2 | 23 |

El PLS-DA presenta una tasa de acierto similar a el PLS-DA que se realizó con el programa "Aspen". La diferencia con respecto a los diferentes métodos se halla en los problemas que tiene en la clasificación con la clase intermedia *"Medium".*

Una vez creados todos los modelos de predicción 100 veces, se ha calculado la media de las tasas de acierto y fallo, obteniéndose para cada método los siguientes resultados (tabla 19):

*Tabla 19. Tasas de acierto según las técnicas de aprendizaje supervisado*

| Método | Tasa de acierto (%) | Tasa de fallo (%) |
|---|---|---|
| RF | 98.1574 | 1.84264 |
| NK | 74.9797 | 25.0203 |
| NB | 83.802 | 16.198 |
| SVM | 92.3655 | 7.63452 |
| Tree | 94.9036 | 5.09645 |
| LS | 83.802 | 16.198 |
| PLSDA | 72.1371 | 27.8629 |

Con el fin de validar estos resultados y comprobar las diferencias que puedan existir significativas entre cada técnica utilizada se ha realizado un ANOVA mediante el programa "*Statgraphics*" con el fin de estudiar el efecto de cada factor. Los factores son los siguientes:

- Método: factor cualitativo que tiene 7 variantes, se trata de todas las técnicas aplicadas de aprendizaje supervisado
- Número de iteración: factor cuantitativo con 100 niveles, se trata del número de iteración para el cálculo de cada tasa de acierto. Este factor se ha utilizado como factor de bloqueo, con el fin de eliminar la variabilidad debida a la iteración

Se pretende estudiar cómo influyen estos dos factores sobre el porcentaje de aciertos para cada iteración según el método utilizado.

*Tabla 20. Análisis de Varianza para Porcentaje de acierto*

| Fuente | Suma de Cuadrados | Grados de libertad | Cuadrado Medio | Razón-F | Valor-P |
|---|---|---|---|---|---|
| EFECTOS PRINCIPALES | | | | | |
| A: Iteración | 0,201229 | 99 | 0,00203262 | 3,33 | 0,0000 |
| B: Método | 533,781 | 6 | 0,889635 | 1458,87 | 0,0000 |
| RESIDUOS | 0,362229 | 594 | 0,000609812 | | |
| TOTAL (CORREGIDO) | 590,127 | 699 | | | |

Según la tabla ANOVA (20), el factor método es estadísticamente significativo sobre la variable dependiente, porcentaje de aciertos, con una probabilidad de error prácticamente nula (Valor-P=0).

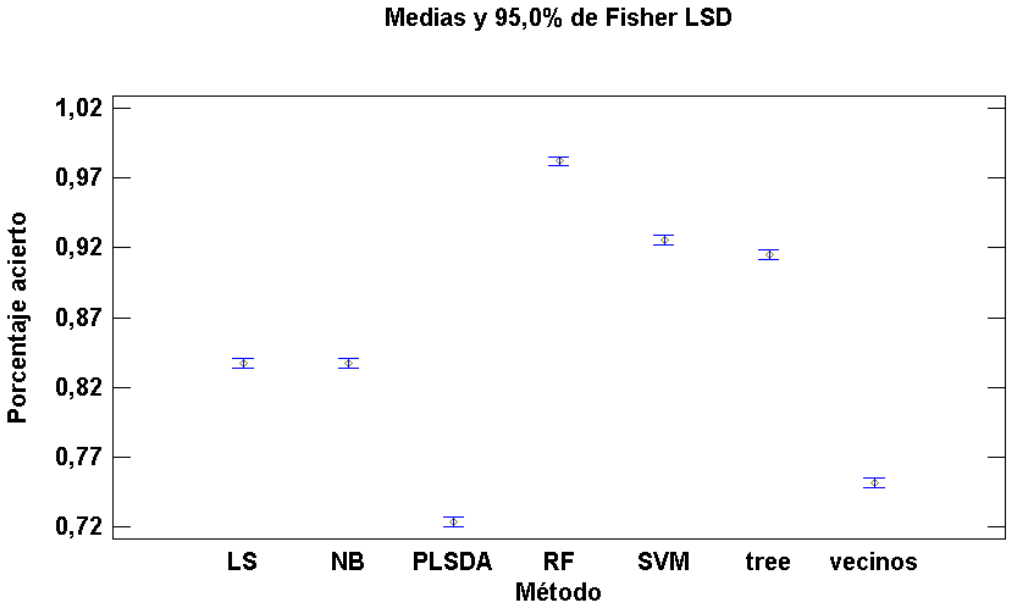



*Figura 24. Intervalos LSD de los diferentes métodos*

A continuación, se ha realizado un gráfico LSD (figura 24) del factor método para comprobar entre qué técnicas existen diferencias. Comprobamos que las técnicas que más diferencias significativas presentan sobre el resto son: Random Forest, árbol de clasificación y Support Vector Machine. También podemos apreciar, que Random Forest (RF), SVM y árboles de clasificación (tree), son las técnicas que más porcentaje de acierto obtienen. Además, se solapan los intervalos LSD de las técnicas Naive Bayes y Laplace Smoothing, así que no existen diferencias significativas con respecto a la media del porcentaje de aciertos en la clasificación de los jamones pastosos.

### 3.3.2Modelos de predicción basados en las variables de calidad y proceso

Tal y como hemos explicado antes, las variables de proceso presentaban una estructura tridimensional. Se ha desdoblado el cubo para poder emplear las herramientas convencionales de análisis transformándolo en una estructura bidireccional. A partir de los análisis exploratorios mediante el PLS, se concluyó que el escalado por bloques de las variables de proceso presentaba mejores resultados. De esta forma, se han autoescalado las variables de calidad y las variables de proceso se han escalado por bloques.

Además, se han balanceado los datos para crear esta nueva base de datos, primero se añadieron todas las variables de proceso en los 7 instantes de tiempo y luego se realizó un remuestreo con remplazamiento hasta completarla. Como en el caso anterior, para la parte de entrenamiento se escogió en cada iteración del Holdout unos 394 datos para la base de datos de entrenamiento y 197 para la parte de validación, igual que en el modelo anterior.

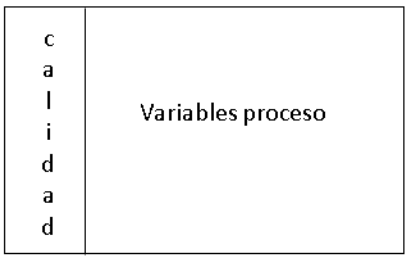



Figura 25. Esquema cualitativo de la estructura de datos utilizada.

La figura 25 muestra un esquema de la estructura de datos utilizada, con un total de 591 jamones (197 de cada clase), 18 variables explicativas de calidad y 56 variables de proceso.

Se presentan a continuación los resultados obtenidos empleando las mismas técnicas que en el anterior modelo así como sus librerías. Se han escogido las matrices de confusión de la última iteración (100) para ilustrar las predicciones realizadas por el modelo con respecto a la clasificación real de los jamones.

Se presentan las siguientes tablas, de la tabla 21 a la 27 presentan ejemplos de las matrices de confusión obtenidas para cada técnica.

*Classification tree*

*Tabla 21. Matriz de confusión iteración 100 Árbol de clasificación*

| real / pred | H | M | L |
|---|---|---|---|
| H | 71 | 4 | 0 |
| M | 0 | 54 | 5 |
| L | 0 | 1 | 62 |

*Random Forest*

*Tabla 22. Matriz de confusión iteración 100 Random Forest*

| real / pred | H | M | L |
|---|---|---|---|
| H | 71 | 1 | 0 |
| M | 0 | 58 | 1 |
| L | 0 | 0 | 66 |

*Nearest Kneighbour*

*Tabla 23. Matriz de confusión iteración 100 Vecinos más próximos*

| real / pred | H | M | L |
|---|---|---|---|
| H | 71 | 8 | 2 |
| M | 0 | 48 | 16 |
| L | 0 | 3 | 49 |

*Naive Bayes*

*Tabla 24. Matriz de confusión iteración 100 Naive Bayes*

| real / pred | H | M | L |
|---|---|---|---|
| H | 71 | 6 | 0 |
| M | 0 | 50 | 13 |
| L | 0 | 3 | 54 |

*Laplace Smoothing*

*Tabla 25. Matriz de confusión iteración 100 Laplace Smoothing*

| pred \ real | H | M | L |
|---|---|---|---|
| H | 71 | 6 | 0 |
| M | 0 | 50 | 13 |
| L | 0 | 3 | 54 |

*Support Vector Machine*

*Tabla 26. Matriz de confusión iteración 100 SVM*

| pred \ real | H | M | L |
|---|---|---|---|
| H | 71 | 4 | 0 |
| M | 0 | 55 | 3 |
| L | 0 | 0 | 64 |

*PLSDA*

*Tabla 27. Matriz de confusión iteración 100 PLS discriminante*

| pred \ real | H | M | L |
|---|---|---|---|
| H | 59 | 19 | 0 |
| M | 12 | 33 | 9 |
| L | 0 | 7 | 58 |

Las predicciones obtenidas en estos modelos son muy similares a los modelos únicamente empleando las variables de calidad. Se muestra en la siguiente tabla 28 la media del porcentaje de aciertos para cada método.

*Tabla 28. Tasas de acierto según las técnicas de aprendizaje supervisado*

| Método | Tasa de acierto (%) | Tasa de fallo (%) |
|---|---|---|
| RF | 98.5736 | 1.4264 |
| NK | 86.5228 | 13.4772 |
| NB | 86.5787 | 13.4213 |
| SVM | 95.736 | 4.26396 |
| Tree | 93.0508 | 6.94924 |
| LS | 86.5787 | 13.4213 |

| | | |
|---|---|---|
| PLSDA | 74.599 | 25.401 |

De la misma forma que en el modelo anterior, se ha realizado un ANOVA para estudiar si existen diferencias significativas entre métodos (tabla 29).

*Tabla 29. Análisis de la varianza*

| Fuente | Suma de Cuadrados | Grados de libertad | Cuadrado Medio | Razón-F | Valor-P |
|---|---|---|---|---|---|
| EFECTOS PRINCIPALES | | | | | |
| A: Iteración | 0,149237 | 99 | 0,00150744 | 2,53 | 0,0000 |
| B: Método | 382,802 | 6 | 0,638003 | 1070,41 | 0,0000 |
| RESIDUOS | 0,354045 | 594 | 0,000596036 | | |
| TOTAL (CORREGIDO) | 43,313 | 699 | | | |

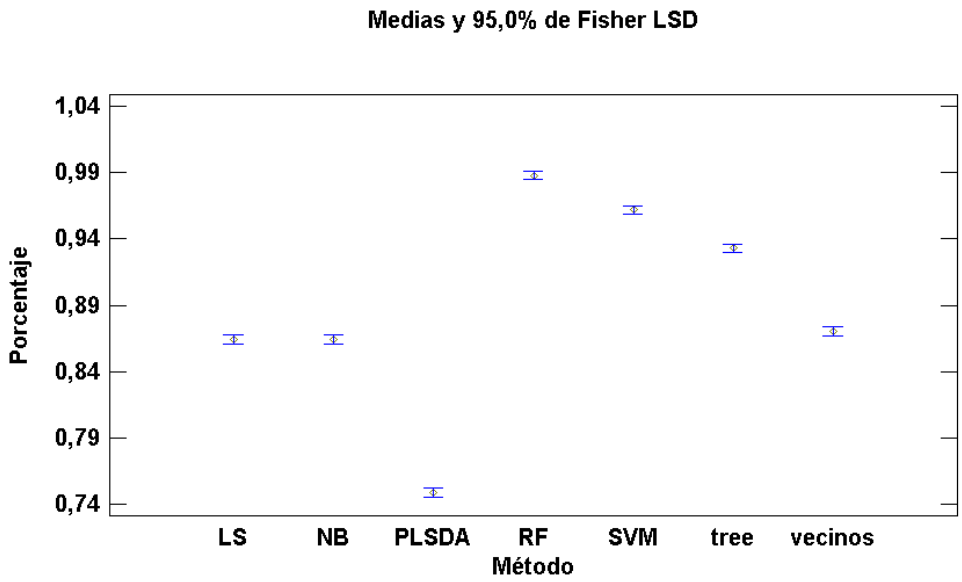



*Figura 26. Intervalos LSD de cada método*

Como podemos observar en el gráfico LSD de la figura 26, Random Forest es el modelo que mejor predice la pastosidad de los jamones seguida de Support Vector Machine. Tampoco existen diferencias entre Naive Bayes y su variante Laplace Smoothing.

## 3.3 Comparación modelos

Uno de los objetivos de este trabajo es demostrar si las variables de proceso y calidad influyen en la pastosidad de los jamones. Se han realizado modelos exploratorios transformando la variable numérica a categórica, se han probado técnicas de aprendizaje supervisado calculando la tasa de acierto de cada modelo de predicción creado. Las variables de proceso provienen del uso experimental de un instrumento de medida costoso como lo es un equipo de ultrasonidos. Este equipo de medición tiene numerosas aplicaciones y una de las que se le quiso dar fue la posible predicción de la pastosidad a partir del espesor de un jamón. Las diferencias entre las tasas de aciertos de todos los modelos obtenidos según el bloque de variable eran muy pocas. En algunos casos, tasas de acierto superiores y en otras inferiores. Por tanto, se va a mostrar un ANOVA en el que se quiere estudiar el efecto que tienen los modelos de variables empleados con sus respectivas técnicas. La hipótesis que se quiere demostrar es la siguiente:

$$H0 : \mu\, Tasas\ de\ acierto_{variables\ de\ calidad} = \mu\, Tasas\ de\ acierto_{variables\ de\ calidad + variables\ de\ proceso}$$

Siendo los elementos del ANOVA los siguientes:

- Factor Bloque Variables: Variable cualitativa con 2 variantes: A (variables de calidad) y B (variables de calidad y proceso)
- Factor Método: Variable cualitativa con 7 variantes: RF (Random Forest), NK (Vecinos más próximos), NB (Naive Bayes), LS (Laplace Smoothing), SVM (Support Vector Machine) y PLSDA (PLS discriminante).
- Porcentaje de aciertos: variable dependiente cuantitativa.

*Tabla 30. Comparación tasas de acierto según el modelo de bloque de variables y su técnica*

| Modelo/Técnica | RF | NK | NB | SVM | Tree | LS | PLSDA |
|---|---|---|---|---|---|---|---|
| Modelos con variables de calidad | 98.2132 | 75.15736 | 83.93401 | 92.49239 | 94.85279 | 83.93401 | 72.71475 |
| Modelos con variables de calidad y proceso | 98.5736 | 86.52284 | 86.57868 | 95.73604 | 93.05076 | 86.57868 | 74.59898 |

*Tabla 31. Análisis de la varianza*

| Fuente | Suma de Cuadrados | Grados de libertad | Cuadrado Medio | Razón-F | Valor-P |
|---|---|---|---|---|---|
| EFECTOS PRINCIPALES | | | | | |
| A: Bloque Variables | 128,004 | 1 | 128,004 | 0,64 | 0,4552 |
| B: Método | 636,674 | 6 | 106,112 | 5,28 | 0,0313 |
| RESIDUOS | 120,532 | 6 | 200,886 | | |
| TOTAL (CORREGIDO) | 770,006 | 13 | | | |

A partir de la tabla 31 sobre el análisis de la varianza de los datos mostrados en la tabla 30, podemos decir que no existen diferencias significativas en el porcentaje de aciertos según el bloque de variable que se utilice, es decir, estructura de datos de proceso y calidad frente a estructuras formadas sólo con datos de calidad. El p-valor es > 5 %, de tal forma que podemos afirmar que no existen diferencias estadísticamente significativas entre utilizar un bloque de variables u otro con un intervalo de confianza del 95%.

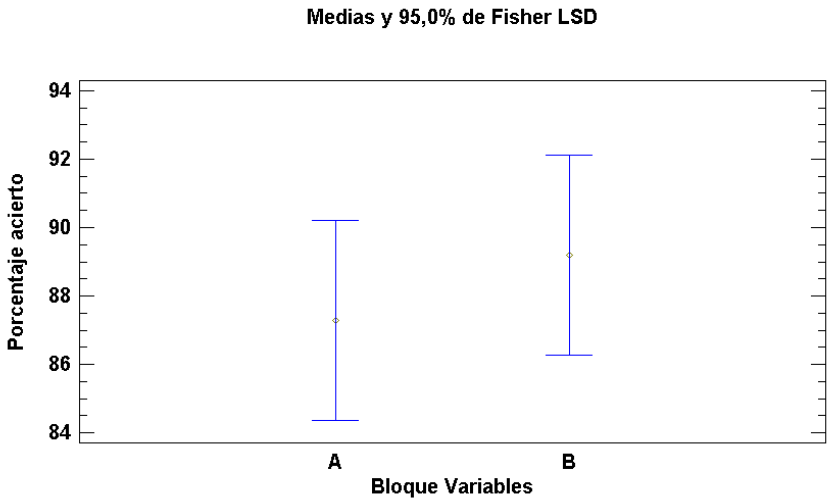



*Figura 27. Intervalos LSD del porcentaje de aciertos según el bloque de variables utilizado*

Efectivamente, la figura 27 muestra que los intervalos LSD se solapan, no hay diferencias en la media según el bloque de variables utilizado.

# CONCLUSIONES

# 4.CONCLUSIONES

En primer lugar, este trabajo ha sido muy beneficioso puesto que se han podido aplicar conocimientos y técnicas aprendidas durante el máster. Se han utilizado programas de análisis y de programación que han desarrollado un aprendizaje a lo largo de la realización de este TFM.

Por otra parte, con respecto a las conclusiones que se han llegado a través de todos los estudios, podemos decir que uno de los principales objetivos era encontrar variables de proceso que pudieran relacionarse con la característica de calidad que hemos estudiado, la pastosidad. Sin embargo, no se han encontrado relaciones directas de predicción con las variables medidas por el equipo de ultrasonidos.

Aunque se hayan encontrado relaciones predictivas muy significativas de las variables de calidad con la pastosidad, tampoco se han encontrado relaciones de las variables de proceso con las de calidad que fueran significativas.

La tecnología ultrasónica puede emplearse para determinar la composición global de un jamón. Sin embargo, en este estudio no se ha encontrado significación con la pastosidad.

Se han realizado modelos de predicción, estudiando entre qué técnicas existen diferencias estadísticamente significativas en la clasificación de la pastosidad. Y aunque hemos explotado la dualidad de la variable respuesta realizando todos los modelos de aprendizaje supervisado con la categórica (clase), consideramos que una variable numérica es mucho más exacta que una clasificación de 3 clases, aporta más información y además presenta muy buena capacidad predictiva en los modelos PLS realizados (pastosidad numérica). Así que consideramos que en un futuro podría investigarse y estudiar los modelos predictivos que se pudieran realizar, así como cambiar la catalogación de la pastosidad de los jamones, de forma que la pastosidad fuera una característica de calidad numérica con un cierto valor representativo.

Hoy en día, cuando se tiene un lote de jamones (p.e 300) que termina el proceso productivo (curación) , se realizan análisis destructivos en cada jamón para obtener muestras que en un futuro serán valoradas por un juez a partir de su criterio subjetivo. Por eso proponemos una metodología basada en los modelos de predicción que han sido propuestos, se presenta un esquema ilustrativo (figura 28):

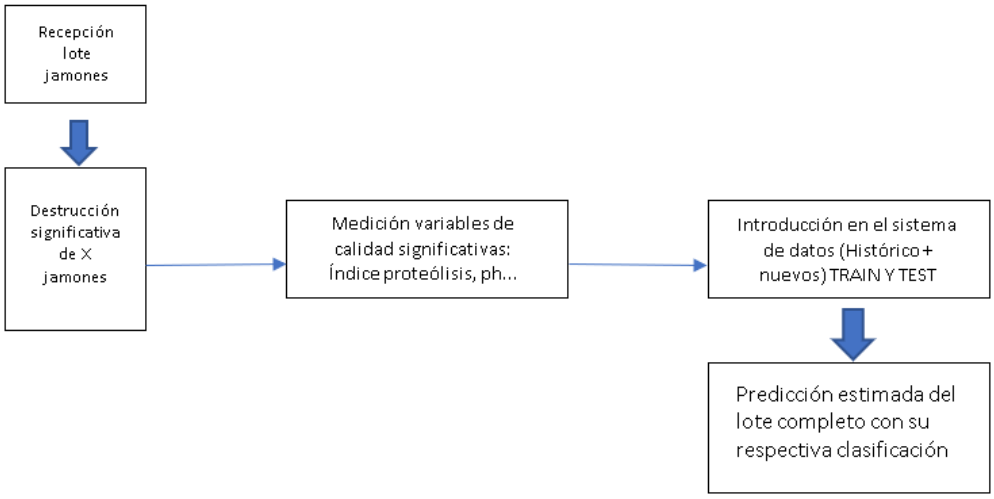



Figura 28. Ilustración básica sobre el procedimiento propuesto

# BIBLIOGRAFÍA

# 5.BIBLIOGRAFÍA

Camacho, J., Picó, J., & Ferrer, A. (2008). Bilinear modelling of batch processes. Part I: Theorical discussion. *Journal of Chemometrics*, 22(5) 299-308.

Cervera Perea, C. (2015). *Trabajo Fin de Grado: Puesta a punto de una metodología para la caracterización de pastosidad en jamón curado mediante ultrasonidos.* Universidad Politécnica de Valencia.

Chaparro, J., Giraldo, B., & Rondón, S. (2013). Evaluación de clasificador Naive Bayes como herramienta de diagnóstico en Unidades de Cuidado Intensivo. *Revista de Tecnología*, 12(2), 87-93.

Dahl, D. B. (2016). Xtable :Export Tables to LaTeX or HTML.

DC, M. (2005). *Design and Analysis of Experiments .*

F, R., B, G., A, S., & K-A, L. C. (2017). mixOmics: An R package for omics feature selection and multiple data integration.

García-Garrido, J., Quiles-Zafra, R., Tapiador, J., & Luque de Castro, M. (1999). Sensory and analytical properties of Spanish dry-cured ham of normal and defective texture. *Food Chemistry*, 67(4), 423-427.

Hastie, T., Tibshirani, R., & Friedman, J. (2001). The elements of statistical learning (vol.1, No.10). Springer series in statistics. New York.

Karatzoglou, A., Smola, A., Hornik, K., & Zeileis, A. (2004). Kernlab-An S4 Package for Kernel Methods in R.

Liaw, A., & Wiener, M. (2002). Classification and Regression by randomForest.

Mevik, B.-H., Wehrens, R., & Hovde Liland, K. (2016). pls: Partial Least Squares and Principal Component Regression.

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., & Leisch, F. (2017). e1071: Misc Functions of the Departament of Statistics, Probability Theory.

Montalbán, J. M. (2005). *Tesis Doctoral: Control Estadístico de Procesos mediante Análisis Multivariante de Imágenes.* Universidad Politécnica de Valencia.

Montgomery, D. C. (2005). *Design and Analysis of Experiments.* John wiley & sons.

Mora Ruiz, M. E. (2015). *Tesis de máster: Influencia de defectos de textura en jamón curado loncheado sobre parámetros ultrasónicos y texturales.* Universidad Politécnica de Valencia.

Mora-Florez, J., Morales-España, G., & Barrera-Cárdenas, R. (2008). Evaluating a k-nearest neighbours-based classifier for locating faulty areas in power systems. *Ingeniería e Investigación*, 28(3), 81-86.

Pedraza, M. P. (2016). *Tesis Doctoral: Caracterización mediante ultrasonidos de señal de los cambios composicionales del jamón curado durante su procesado.* Universidad Politécnica de Valencia.

R core, T. (2018). A language and environment for statistical computing.

Rodrigo, J. A. (2017). *Árboles de predicción: bagging, random forest, boosting y C5.0.* Obtenido de Rpubs.com: https://rpubs.com/Joaquin_AR/255596

Ruiz, M. E. (2015). *Tesis de master: Influencia de defectos de textura del jamón sobre parámetros ultrasónicos.* Universidad Politécnica de Valencia.

Sanchez, G. (2013). DiscriMiner: Tools of the Trade for Discriminant Analysis.

Thernau, T., & Atikson, B. (2018). rpart: Recursive Partioning and Regression Trees.

Thompson, C., & Shure, L. (1995). *Image Processing Toolbox: For Use with MATLAB*. Obtenido de Mathworks

Torgo, L. (2010). Data Mining with R, learning with case studies Chapman and Hall/CRC.

Vargas, J., Conde, B., Paccapelo, V., & Zingaaretti, L. (2012). *Máquinas de soporte vectorial : Metodología y aplicación en r*. Obtenido de http://conferencias. unc. edu. ar/index. php/xclatse/clatse2012/paper/view/265 [Ultimo acceso: 26 de abril 2013].

Venables, W. N., & Ripley, B. (2002). Tree-based methods. In Modern Applied Statistics with S. *Springer*, 251-269.

Westerhuis, J., Kourti, T., & MacGregor, J. F. (1999). Comparing alternative approaches for multivariate statistical analysis of batch process data. *Journal Chemometrics: A Journal of the Chemometrics Society,*, 13(3-4), 397-413.

Wickham, H., & Bryan, J. (2018). readxl: Read Excel Files.

Wold, S. E. (1987). Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3), 37-52.

Wold, S., Sjöström, M., & Eriksson, L. (2001). PLS-regression: a basic tool of chemometrics. *Chemometrics and intelligent laboratory systems*, 58(2), 109-130.

# ANEXOS

# 6.ANEXOS

## 6.1 Código aprendizaje supervisado calidad

```
library(ade4)
library(ggplot2)
library(MASS)
library(lattice)
library(mixOmics)
library(xtable)
library(rpart)
library(readxl)
library (e1071)
library(randomForest)
library(class)
library("kernlab")
library(grid)
library(DMwR)
library(grid)
library(MASS)
library(FactoMineR)
library(DiscriMiner)

Datos<- read_excel("Datos balanceados calidad R.xlsx")
head(Datos)

#escalamos#
clase=Datos[,18]
Datos=scale(Datos[,c(-18)],center=TRUE,scale=TRUE) #centramos y escala
mos a varianza unitaria#

Datos=data.frame(Datos[,c(-1,-2)],clase)
View(Datos)

clase=factor(Datos[,16])
```

```
tasaaciertoRF=NA

tasaaciertoNK=NA

tasaaciertoNB=NA

tasaaciertoLS=NA

tasaaciertosSVM=NA

tasaaciertostree=NA

tasaaciertoPLSDA=NA


# Random forest ----------------------------------------------------
-----


r.f=randomForest(factor(Datos$clase)~ ., data=Datos,mtry=3,method="cla
ss",importance=TRUE)


print(r.f)

importance(r.f)


#el var plot nos dice que variables son las mas importantes


varImpPlot(r.f, main="",col="#CC00FFFF")


par(mfrow=c(1,1))


varImpPlot(r.f, main="",col="blue", type=1)

varImpPlot(r.f, main="",col="red", type=2)


#la viscosidad saliva es la variable mas importante#


#tratamos de utilizar la tenica para predecir#


for (i in 1:100){


  Datosmodel=sample(1:591,394, FALSE) #seleccionamos el 66,66 por cien
to para entrenamiento#

  tr=Datos[Datosmodel,]

  ts=Datos[-Datosmodel,]


  m=randomForest(factor(tr$clase)~ ., tr)
```

```
  ps=predict(m, ts)

  tabRF=table(ps, ts$clase)

  tabRF

  tasaaciertoRF[i]=sum(tabRF[row(tabRF)==col(tabRF)])/sum(tabRF) #tasa
de acierto #

  tasafalloRF=sum(tabRF[row(tabRF)!=col(tabRF)])/sum(tabRF) #tasa de f
allo#

  tasafalloRF

  RF=c(tasaaciertoRF,tasafalloRF)

  RF

  names(RF)=c("tasa acierto","tasa fallo")

  RF


  # nearest kneighbour ------------------------------------------------
-------




  vecino=knn(tr[,-16], ts[,-16], factor(tr$clase), k = 14, prob = TRUE
)


  summary(vecino)


  tabNK=table(vecino,factor(ts$clase))

  tabNK

  tasaaciertoNK[i]=sum(tabNK[row(tabNK)==col(tabNK)])/sum(tabNK)

  tasafalloNK=sum(tabNK[row(tabNK)!=col(tabNK)])/sum(tabNK)


  NK=c(tasaaciertoNK,tasafalloNK)

  NK

  names(NK)=c("tasa acierto","tasa fallo")

  NK

  # Naive Bayes -------------------------------------------------------
-------



  nbayes=naiveBayes(factor(clase)~ ., data=tr)

  summary(nbayes)
```

```
  pred=predict(nbayes, ts[,-16])
  tabNB=table(pred, ts$clase)
  tabNB
  tasaaciertoNB[i]=sum(tabNB[row(tabNB)==col(tabNB)])/sum(tabNB)
  tasafalloNB=sum(tabNB[row(tabNB)!=col(tabNB)])/sum(tabNB)


  NB=c(tasaaciertoNB,tasafalloNB)
  NB
  names(NB)=c("tasa acierto","tasa fallo")
  NB



  # SVM ------------------------------------------------------------
-------



  svp=ksvm(factor(clase)~ ., data=tr, type = "C-svc", kernel = "rbfdot
",kpar = "automatic")
  summary(svp)
  pred=predict(svp, ts[,-16])
  tabSVM=table(pred, ts$clase)
  tabSVM
  tasaaciertosSVM[i]=sum(tabSVM[row(tabSVM)==col(tabSVM)])/sum(tabSVM)
  tasaaciertosSVM




  # Classification Tree ------------------------------------------
-------



  treefull=rpart(clase~ .,data=tr,method="class", cp=0.005)
  plotcp(treefull)
  printcp(treefull)
  prettyTree(treefull)
  tree2=prune.rpart(treefull,0.033)
  prettyTree(tree2)
```

```
  pred=predict(tree2, ts[,-16],type = "class")

  tabtree=table(pred, ts[,16])

  tabtree

  tasaaciertostree[i]=sum(tabtree[row(tabtree)==col(tabtree)])/sum(tab
tree)

  tasaaciertostree



  # Laplace Smoothing ------------------------------------------------
-------


  model=naiveBayes(factor(clase)~ ., data=tr, laplace = 3)

  pred=predict(model, ts[,-16])

  tabLS=table(pred,ts[,16])

  tabLS

  tasaaciertoLS[i]=sum(tabLS[row(tabLS)==col(tabLS)])/sum(tabLS)



# PLSDA ------------------------------------------------------------------
-------


  #n=plsDA(tr[,c(-16)], tr$clase, autosel = FALSE, comps = 4,

         # cv = "LOO", k = NULL, retain.models = FALSE)


  #n=plsDA(Datos[,c(-16)], Datos$clase, autosel = FALSE, comps = 4,val
idation = "learntest",

       # learn =tr[,c(-16)], test =ts,

        #cv = "LOO", k = NULL, retain.models = FALSE)


  #tabPLSDA=n$confusion

  #tasaaciertoPLSDA[i]=sum(tabPLSDA[row(tabPLSDA)==col(tabPLSDA)])/sum
(tabPLSDA)

  #graf=plot(n)



  #learning=sample(1:591,394, FALSE)

  #esting=sample(1:591,197, FALSE)

  #my_pls3=plsDA(Datos[,1:15],Datos$clase,validation="learntest",learn
=learning,test=testing)
```

```
  X=tr[,c(-16)]
  Y=tr$clase
  m=plsda(X,Y,ncomp=4,scale=FALSE)

  pred=predict(m, ts[,-16])
  predictions=pred$class$max.dist[,4]
  tabPLSDA=table(predictions,ts[,16])
  tabPLSDA
  tasaaciertoPLSDA[i]=sum(tabPLSDA[row(tabPLSDA)==col(tabPLSDA)])/sum(
tabPLSDA)

}

tasa=cbind(tasaaciertoRF,tasaaciertoNB,tasaaciertoNK,tasaaciertosSVM,t
asaaciertostree,tasaaciertoLS,tasaaciertoPLSDA)
mediatasa=apply(tasa,2,mean)
mediatasa
tasa.int=apply(tasa,2, quantile, probs=c(0.025,0.975))
tasa.int

vector.tasa=as.vector(tasa)

cbind(vector.tasa[1:100], tasa[,1])

metodo<-rep(c("RF","NK","NB","SVM","tree","LS","PLSDA"), each=100)
plot(factor(metodo),vector.tasa)

abline(0.85,0, col="red")
abline(0.89,0, col="blue")
abline(0.95,0, col="green")

#voy a comprobar si hay diferencias entre RF y NK que parecen que no p
resenten diferencias, asi comprobamos#
t.test(tasa[metodo=="NB"], tasa[metodo=="LS"], alternative = c("two.si
ded", "less", "greater"), paired = TRUE, var.equal = FALSE, conf.level
= 0.95)
```

## 6.2 Código aprendizaje supervisado calidad y proceso

```
library(ade4) #PCA
library(ggplot2)
library(MASS)
library(lattice)
library(mixOmics)
library(xtable)
library(C50)
library(rpart)
library(readxl)
library (e1071)
library(randomForest)
library(class)
library("kernlab")
library(grid)
library(DMwR)
library(grid)


Datos<- read_excel("Datos balanceados calidad y proceso R.xlsx")


head(Datos)


#escalamos#
clase=Datos[,74]
proceso=Datos[,c(18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35
,36,37,38,39,40,
               41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59
,60,61,62,63,64,
               65,66,67,68,69,70,71,72,73)]


Datos=scale(Datos[,c(-18,-19,-20,-21,-22,-23,-24,-25,-26,-27,-28,-29,-
30,-31,-32,-33,-34,-35,-36,-37,-38,-39,-40,
                     -41,-42,-43,-44,-45,-46,-47,-48,-49,-50,-51,-52,-
53,-54,-55,-56,-57,-58,-59,-60,-61,-62,-63,-64,
                     -65,-66,-67,-68,-69,-70,-71,-72,-73,-74)],center=
TRUE,scale=TRUE) #centramos y escalamos a varianza unitaria#
```

```
Datos=data.frame(Datos[,c(-1,-2)],proceso,clase)
View(Datos)



tasaaciertoRF=NA
tasaaciertoNK=NA
tasaaciertoNB=NA
tasaaciertoLS=NA
tasaaciertosSVM=NA
tasaaciertostree=NA
tasaaciertoPLSDA=NA



# Random forest ------------------------------------------------------
-----


r.f=randomForest(factor(Datos$clase)~ ., data=Datos,mtry=3,method="cla
ss",importance=TRUE)
print(r.f)
importance(r.f)


#el var plot nos dice que variables son las mas importantes


colors()[1:20] #miro los colores#
rainbow(10)


varImpPlot(r.f, main="",col="#CC00FFFF")


par(mfrow=c(1,1))


varImpPlot(r.f, main="",col="blue", type=1)
varImpPlot(r.f, main="",col="blue", type=2)


#la viscosidad saliva es la variable mas importante#


#tratamos de utilizar la tenica para predecir#


for (i in 1:100){
```

```
  Datosmodel=sample(1:591,394, FALSE) #seleccionamos el 66,66 por cien
to para entrenamiento#
  tr=Datos[Datosmodel,]
  ts=Datos[-Datosmodel,]


  m=randomForest(factor(tr$clase)~ ., tr)
  ps=predict(m, ts)
  tabRF=table(ps, ts$clase)
  tabRF
  tasaaciertoRF[i]=sum(tabRF[row(tabRF)==col(tabRF)])/sum(tabRF) #tasa
de acierto #
  tasafalloRF=sum(tabRF[row(tabRF)!=col(tabRF)])/sum(tabRF) #tasa de f
allo#
  tasafalloRF
  RF=c(tasaaciertoRF,tasafalloRF)
  RF
  names(RF)=c("tasa acierto","tasa fallo")
  RF


  # nearest kneighbour ------------------------------------------------
-------


  vecino=knn(tr[,-72], ts[,-72], factor(tr$clase), k = 3, prob = TRUE)
  plot(vecino)
  summary(vecino)


  tabNK=table(vecino,factor(ts$clase))
  tabNK
  tasaaciertoNK[i]=sum(tabNK[row(tabNK)==col(tabNK)])/sum(tabNK)
  tasafalloNK=sum(tabNK[row(tabNK)!=col(tabNK)])/sum(tabNK)


  NK=c(tasaaciertoNK,tasafalloNK)
  NK
  names(NK)=c("tasa acierto","tasa fallo")
  NK
  # Naive Bayes -------------------------------------------------------
-------
```

```
  nbayes=naiveBayes(factor(clase)~ ., data=tr)
  summary(nbayes)

  pred=predict(nbayes, ts[,-72])
  tabNB=table(pred, ts$clase)
  tabNB
  tasaaciertoNB[i]=sum(tabNB[row(tabNB)==col(tabNB)])/sum(tabNB)
  tasafalloNB=sum(tabNB[row(tabNB)!=col(tabNB)])/sum(tabNB)

  NB=c(tasaaciertoNB,tasafalloNB)
  NB
  names(NB)=c("tasa acierto","tasa fallo")
  NB



  # SVM ---------------------------------------------------------------
-------
  svp=ksvm(factor(clase)~ ., data=tr, type = "C-svc", kernel = "rbfdot
",kpar = "automatic")
  summary(svp)
  pred=predict(svp, ts[,-72])
  tabSVM=table(pred, ts$clase)
  tabSVM
  tasaaciertosSVM[i]=sum(tabSVM[row(tabSVM)==col(tabSVM)])/sum(tabSVM)
  tasaaciertosSVM



  # Classification Tree -----------------------------------------------
-------



  treefull=rpart(clase~ .,data=tr,method="class", cp=0.001)
  plotcp(treefull)
  printcp(treefull)
  prettyTree(treefull)
  tree2=prune.rpart(treefull,0.01)
  prettyTree(tree2)
```

```
  pred=predict(tree2, ts[,-72],type = "class")
  tabtree=table(pred, ts[,72])
  tabtree
  tasaaciertostree[i]=sum(tabtree[row(tabtree)==col(tabtree)])/sum(tab
tree)
  tasaaciertostree



  # Laplace Smoothing ------------------------------------------------
-------


  model=naiveBayes(factor(clase)~ ., data=tr, laplace = 3)
  pred=predict(model, ts[,-72])
  tabLS=table(pred,ts[,72])
  tabLS
  tasaaciertoLS[i]=sum(tabLS[row(tabLS)==col(tabLS)])/sum(tabLS)



# PLSDA ------------------------------------------------------------
-------


  X=tr[,c(-72)]
  Y=tr$clase
  m=plsda(X,Y,ncomp=4,scale=FALSE)


  pred=predict(m, ts[,-72])
  predictions=pred$class$max.dist[,4]
  tabPLSDA=table(predictions,ts[,72])
  tabPLSDA
  tasaaciertoPLSDA[i]=sum(tabPLSDA[row(tabPLSDA)==col(tabPLSDA)])/sum(
tabPLSDA)




}


tasa=cbind(tasaaciertoRF,tasaaciertoNB,tasaaciertoNK,tasaaciertosSVM,t
asaaciertostree,tasaaciertoLS,tasaaciertoPLSDA)
```

```
mediatasa=apply(tasa,2,mean)

mediatasa

tasa.int=apply(tasa,2, quantile, probs=c(0.025,0.975))

tasa.int


vector.tasa=as.vector(tasa)


cbind(vector.tasa[1:100], tasa[,1])


metodo<-rep(c("RF","NK","NB","SVM","tree","LS","PLSDA"), each=100) #no
puedo hacer el grafico, no se pq#

plot(factor(metodo),vector.tasa)


abline(0.87,0, col="red")

abline(0.9,0, col="blue")

abline(0.97,0, col="green")


#voy a comprobar si hay diferencias entre RF y NK que parecen que no p
resenten diferencias, asi comprobamos#

t.test(tasa[metodo=="RF"], tasa[metodo=="LS"], alternative = c("two.si
ded", "less", "greater"), paired = TRUE, var.equal = FALSE, conf.level
= 0.95)



write.csv(tasaaciertoRF, file="RF.csv")

write.csv(tasaaciertoNK, file="NK.csv")

write.csv(tasaaciertoNB, file="NB.csv")

write.csv(tasaaciertosSVM, file="SVM.csv")

write.csv(tasaaciertoLS, file="LS.csv")

write.csv(tasaaciertoPLSDA, file="PLSDA.csv")

write.csv(tasaaciertostree, file="tree.csv")


citation()

citation("pkgname")
```