



**UNIVERSIDAD
DE ANTIOQUIA**

**EVALUACIÓN DE LA SEVERIDAD MOTORA DE
PACIENTES CON PARKINSON POR MEDIO DE SENSORES
DE PROFUNDIDAD**

Autor:

Santiago Andres Moreno Acevedo

Universidad de Antioquia

Facultad de Ingeniería, Departamento de Ingeniería Electrónica y
Telecomunicaciones

Medellín, Colombia

2021



EVALUACIÓN DE LA SEVERIDAD MOTORA DE PACIENTES CON PARKINSON POR MEDIO DE SENSORES DE PROFUNDIDAD

Santiago Andres Moreno Acevedo

Trabajo de grado como requisito para optar al título de:
Ingeniero Electrónico

Asesores:

Prof. Dr.-Ing. Juan Rafael Orozco Arroyave
Juan Camilo Vásquez Correa MSc.

Línea de investigación:

Procesamiento Digital de Señales

Grupo de investigación:

Grupo de investigación en Telecomunicaciones Aplicadas GITA

Universidad de Antioquia
Facultad de Ingeniería, Departamento de Ingeniería Electrónica y
Telecomunicaciones
Medellín, Colombia
2021.

Tabla de Contenido

Resumen	4
1. Introducción	5
2. Objetivos	7
2.1. Objetivo general	7
2.2. Objetivos específicos	7
3. Marco teórico	7
3.1. Seguimiento del talón	7
3.1.1. Tracker	7
3.1.2. Flujo óptico	7
3.2. Extracción de características	8
3.2.1. Jitter	8
3.2.2. Shimmer	9
3.2.3. Energia	9
3.2.4. Análisis de componentes principales (PCA)	9
3.3. Validación cruzada	11
3.4. Clasificación	12
3.4.1. Máquinas de soporte vectorial (SVM)	12
3.4.2. Bosques aleatorios (RF)	14
3.4.3. Curva ROC	15
3.4.4. Matriz de confusión	16
3.5. Regresión	17
3.5.1. Regresión por vectores de soporte (SVR)	17
3.5.2. Regresión por bosques aleatorios (RF para regresión)	18
3.5.3. Coeficiente de correlación de rangos de Spearman	18
3.5.4. Error medio Absoluto (MAE)	18
4. Metodología	18
4.1. Base de datos	18
4.2. Seguimiento del talón	19
4.3. Extracción de características	19
4.4. Evaluación	19
4.4.1. Clasificación	21
4.4.2. Regresión	22
5. Resultados y análisis	24
5.1. Seguimiento del talón	24
5.1.1. Manual	24
5.1.2. Automática	24
5.2. Extracción de características	25
5.3. Evaluación	27
5.3.1. SVM y RF	27
5.3.2. SVR y RF para regresión	34
6. Conclusiones	40
7. Líneas Futuras	41
Referencias Bibliográficas	42

Resumen

La enfermedad de Parkinson ha sido altamente estudiada por el grupo GITA perteneciente a la Universidad de Antioquia. Para esto se han utilizado diferentes tipos de señales, entre las cuales se encuentran las señales de audio, de escritura, de marcha, entre otras. La severidad de esta enfermedad es medida por medio del índice MDS-UPDRS. En este trabajo utilizamos una sub-escala de dicho ítem enfocada en el estado motor de los miembros inferiores, llamándose UPDRS-Lower-Limbs.

En el presente trabajo se realizó un estudio de los trastornos de movimiento generados por la enfermedad de Parkinson enfocándose en la motricidad del movimiento en los miembros inferiores. Para esto se propone utilizar señales de movimiento captadas por medio de sensores de profundidad. A partir de estas señales se extraen como características el Jitter, el Shimmer y la energía del movimiento. Con las características se busca evaluar el estado de severidad en los miembros inferiores de los pacientes. Para esto se consideran 2 enfoques. El primero es basado en clasificación. Se dividen las muestras en 2 grupos (Severidad baja y severidad alta) y se realiza la clasificación binaria con estos grupos. Para este enfoque se usan las técnicas de máquinas de soporte Vectorial y bosques aleatorios. El segundo enfoque está basado en regresión. Se genera un modelo que se adapte a los datos e intente predecir el índice. Para este método se usaron las técnicas regresión por vectores de soporte y regresión por bosques aleatorios.

Los resultados indican que en clasificación se puede obtener una exactitud de hasta 73%, y en regresión se puede obtener un coeficiente de correlación de hasta 0.51 con un error medio absoluto de 10.98. En este trabajo se encontró los mejores parámetros para las diferentes técnicas de evaluación y se realizaron comparaciones entre las diferentes técnicas y métodos.

Introducción

El uso de sensores de profundidad para el seguimiento de objetos es un método relativamente nuevo. Desde la llegada del sensor Kinect de Microsoft, el uso de este tipo de sensores se ha popularizado desde el análisis del funcionamiento del sensor [15] hasta su aplicación en el seguimiento de objetos [13][14][16]. La llegada de este sensor abrió nuevas puertas en el campo de la visión artificial gracias al uso del principio de triangulación y a la inclusión de la componente Z (profundidad) de una imagen.

Su utilidad en el campo de la visión artificial ha sido investigada en múltiples aplicaciones desde aplicaciones en bioingeniería como el seguimiento de los brazos de un cirujano [14], hasta el análisis de marcha por medio de puntos clave de la estructura [13][15].

El campo que este tipo de sensores deja a investigar es bastante amplio. Gracias a la posibilidad de realizar el seguimiento del movimiento de las personas es posible usar este tipo de sensores en diferentes áreas. Un campo de uso recientemente explorado es el análisis de pacientes con la enfermedad de Parkinson. Para esto se ha estudiado el movimiento de las manos y de los dedos usando el sensor de profundidad Kinect para obtener características como la velocidad, aceleración, frecuencia y amplitud del movimiento logrando una precisión de entre 75% y 100% de aciertos según la tarea realizada utilizando máquinas de soporte vectorial con Kernel RBF [17]. Una particularidad de los estudios realizados es que cuentan con un robusto algoritmo de seguimiento del objeto de interés, dichos algoritmos presentan detección de errores por índices como la solidez, el área normalizada y la estabilidad [17]. También se han realizado estudios con movimientos de mayor magnitud como el movimiento de la marcha [24][25][26]. El estudio realizado en el Instituto de Tecnología Química en Praga [25], el cuál utiliza el sensor Kinect para el análisis de marcha, en donde se obtiene la longitud del paso y la velocidad de 18 pacientes con la enfermedad de Parkinson y 18 controles sanos, consiguiendo un acierto de más del 90% en la clasificación de estos [25][26].

La llegada del Kinect como un sensor de bajo costo propone su uso sobre otros métodos. En el reino unido se hizo un estudio donde se comparó la precisión del sensor Kinect con el sistema de movimiento Vicon de Oxford [19], a la hora de realizar el seguimiento del movimiento de pacientes con la enfermedad de Parkinson en donde se concluye que el sensor Kinect realiza un buen seguimiento en movimientos grandes, pero tiene grandes fallos en movimientos pequeños [19].

También se ha investigado acerca de la hipomimia generada por la enfermedad de Parkinson en donde se usa un software comercial para el seguimiento de los puntos de interés del rostro y con esto brinda información

sobre la intensidad de 51 unidades de acción faciales, llegando a obtener valores de 90% de área bajo la curva ROC [18].

La enfermedad de Parkinson es una enfermedad neuro-degenerativa que afecta el sistema motor del paciente generando síntomas como, temblor en las manos, los brazos, las piernas, la mandíbula y la cara; rigidez en los brazos, las piernas y el tronco; lentitud de los movimientos y problemas de equilibrio y coordinación [21]. Este desorden ha sido altamente estudiado por el grupo de investigación GITA (Grupo de investigación en telecomunicaciones aplicadas) perteneciente a la Universidad de Antioquia. En estos estudios realizados se ha trabajado con diferentes tipos de señales con diferentes enfoques, algunos ejemplos de esto son las señales de audio para clasificación de pacientes sanos y pacientes con la enfermedad de Parkinson [1][23], señales de escritura para la evaluación de diferentes etapas de la enfermedad de Parkinson por medio del índice MDS-UPDRS [7] de un paciente con Parkinson [2][22], entre otros.

En el grupo de investigación se dispone de una base de datos [12] que consiste en 30 videos de personas con la enfermedad de Parkinson de las cuales se tiene el índice UPDRS-Lower-Limbs realizando un movimiento repetitivo tocando punta y talón con el pie. Debido a los síntomas que produce esta enfermedad, este movimiento se ve afectado según el grado de avance del Parkinson. Haciendo uso de medios computacionales es posible obtener información útil de este movimiento para realizar estudios sobre la severidad motora del paciente. Esta información podría ser utilizada en diferentes algoritmos para realizar predicciones sobre el índice UPDRS-Lower-Limbs, para esto es necesario realizar una buena extracción de información con el fin de que esta sea lo más concisa posible.

Una forma de captar este movimiento es por medio de sensores de profundidad como el Kinect. Este genera una imagen en escala de grises en donde la tonalidad de los grises da información sobre la profundidad de los objetos. Así se generan grabaciones de diferentes movimientos de los pacientes. La información se extrae procesando las imágenes que el sensor Kinect capte obteniendo características que representen la información relevante del movimiento. Esto resulta en un dataset para generar modelos que ayuden en la evaluación del índice UPDRS-Lower-Limbs de pacientes con Parkinson.

Este estudio brinda información sobre el efecto de la enfermedad de Parkinson en el movimiento de una persona, y aporta una perspectiva de análisis computacional, dejando una base para el desarrollo de aplicaciones que puedan ayudar al diagnóstico y tratamiento de esta enfermedad.

1. Objetivos

1.1. Objetivo general

Realizar la evaluación de la severidad motora en pacientes diagnosticados con la enfermedad de Parkinson por medio características extraídas de una base de datos de vídeos tomados con el sensor de profundidad Kinect, con el fin de entrenar modelos que realicen la evaluación por medio clasificación y regresión del índice UPDRS-Lower-Limbs.

1.2. Objetivos específicos

- Realizar seguimiento del talón en los vídeos tomados con el sensor Kinect a personas con la enfermedad de Parkinson, para poder extraer características.
- Procesar los datos para obtener características como el Jitter, Shimmer y la energía del movimiento, para poder utilizarlos en el entrenamiento de modelos de evaluación.
- Utilizar las características para entrenar modelos que realicen evaluación de la severidad motora por medio de clasificación del índice UPDRS-Lower-Limbs y otros que realicen la evaluación por medio de regresión del mismo índice.
- Evaluar el desempeño de los modelos previamente entrenados, con los índices de desempeño indicados para cada modelo.

2. Marco Teórico

2.1. Seguimiento del talón

2.1.1. Tracker

Tracker es un programa libre para análisis de video y modelamiento desarrollado en el framework de Java Open Source Physics (OSP). Es de gran utilidad para realizar seguimiento de partículas, analizar fenómenos físicos en vídeos y realizar un modelamiento de estos.

Este programa entregará los datos del movimiento del talón de la persona, como el punto de máxima altura y el punto en el que el talón toca el suelo.

2.1.2. Flujo óptico

El flujo óptico es el patrón del movimiento aparente de los objetos. En visión artificial es un algoritmo que detecta el movimiento entre 2 imágenes consecutivas por medio de las derivadas espaciales y de la derivada temporal de la imagen.

Se supone una imagen $I(x, y)$ la cual tiene un pequeño movimiento u en X y v en Y , dando como resultado la imagen $H(x, y)$.

Entonces se puede decir que:

La imagen I con el movimiento aplicado menos la imagen resultante H , es igual a 0 (Ecuación 1).

$$0 = I(x + u, y + v) - H(x, y) \quad (1)$$

Sea

$$I_x = \frac{\partial I}{\partial x} \quad (2)$$

Se puede expresar el movimiento como el cambio en las componentes X y Y. Este cambio en el límite tiende a 0, entonces se expresa como derivada (Ecuación 3).

$$0 \approx I(x, y) + I_x u + I_y v - H(x, y) \quad (3)$$

Así la resta de la imagen original I y la imagen con el movimiento aplicado H es el cambio en las componentes respecto al tiempo. Como se toman 2 instantes seguidos se expresa como derivada temporal (Ecuación 4 y 5).

$$0 \approx (I(x, y) - H(x, y)) + I_x u + I_y v \quad (4)$$

$$0 \approx I_t + I_x u + I_y v \quad (5)$$

$$0 \approx I_t + \nabla I \cdot [u \ v] \quad (6)$$

Como en el límite u y v tienden a 0, se tiene una equivalencia a (Ecuación 7):

$$0 = I_t + \nabla I \cdot \left[\frac{\partial x}{\partial t} \ \frac{\partial y}{\partial t} \right] \quad (7)$$

$$\nabla I \cdot \vec{V} = -I_t \quad (8) \text{ Ecuación del flujo óptico}$$

En la ecuación 8, ecuación del flujo óptico, se busca encontrar el vector \vec{V} que supone el movimiento en X y Y de la imagen.

2.2. Extracción de características

2.2.1. Jitter [3]

- Jitter (Absoluto): Es la variación ciclo a ciclo de la frecuencia fundamental de la señal de interés, es el promedio del valor absoluto de la diferencia de la frecuencia fundamental entre periodos consecutivos.

$$Jitter(absolute) = \frac{1}{N-1} \sum_{i=0}^{N-1} |T_i - T_{i-1}| \quad (9)$$

Donde T_i es el valor de cada periodo extraído, y N es el número de periodos analizados.

- Jitter (Relativo): Es el promedio del valor absoluto de la diferencia entre periodos consecutivos, dividido por el periodo promedio.

$$Jitter(\text{relativo}) = \frac{\frac{1}{N-1} \sum_{i=0}^{N-1} |T_i - T_{i-1}|}{\frac{1}{N} \sum_{i=0}^{N-1} T_i} \quad (10)$$

2.2.2. Shimmer [3]

- Shimmer (dB): Es expresado como la variabilidad de la amplitud pico a pico en decibeles. El promedio del valor absoluto del logaritmo en base 10 de la diferencia entre las amplitudes de periodos consecutivos.

$$Shimmer \text{ (dB)} = \frac{1}{N-1} \sum_{i=0}^{N-1} \left| 20 \log \left(\frac{A_{i+1}}{A_i} \right) \right| \quad (11)$$

Donde A_i es la amplitud pico a pico de cada periodo y N es el número de periodos Analizados.

- Shimmer (relativo): Es definido como el promedio del valor absoluto de la diferencia entre las amplitudes pico a pico de periodos consecutivos, dividido por la amplitud promedio, expresado como porcentaje.

$$Shimmer \text{ (relativo)} = \frac{\frac{1}{N-1} \sum_{i=0}^{N-1} |A_i - A_{i-1}|}{\frac{1}{N} \sum_{i=0}^{N-1} A_i} \quad (12)$$

2.2.3. Energía

- Es la energía promedio que conlleva el movimiento físico del pie al subir y bajar el talón.

$$E = \frac{1}{N} \sum_{i=0}^{N-1} E_i \quad (13)$$

- Donde E_i es la energía en cada tramo del movimiento y es calculada con la velocidad de cada tramo v_i estas se expresan como:

$$E_i = \frac{1}{2} v_i^2 \quad (14)$$

$$v_i = \frac{d_i - d_{i+1}}{t_i - t_{i+1}} \quad (15)$$

t_i y d_i es el momento y la posición del talón cuando se encuentra en su posición de máxima o mínima altura.

2.2.4. Análisis de componentes principales (PCA) [9]

El análisis de componentes principales o PCA por sus siglas en inglés, es una técnica de reducción de dimensionalidad ampliamente utilizada en el campo del aprendizaje de máquina.

Esta técnica toma un conjunto de variables y busca reducir su dimensión sin generar una gran pérdida de información. Para esto la técnica se apoya en los conceptos de transformación lineal y valores propios para generar nuevas variables (componentes) maximizando la varianza de estas componentes. En la figura 1 se muestra gráficamente la búsqueda de componentes (Y1 y Y2).

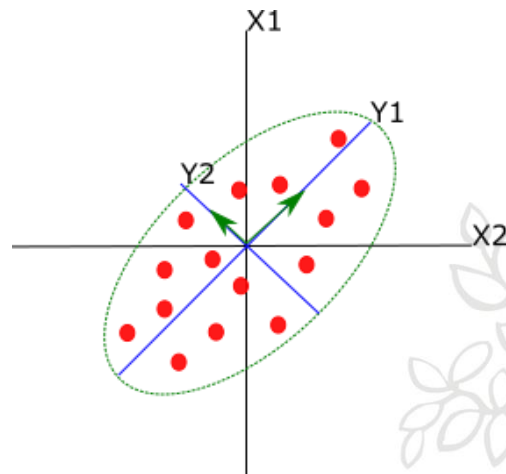


Figura 1. Grafica PCA con 2 componentes

Se supone un conjunto de datos $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3, \dots, \mathbf{X}_m\} \forall X_i \in R^n$, lo cuales tienen media 0 y desviación estándar 1.

Aplicando una transformación lineal a este conjunto se obtiene el conjunto Y.

$$Y_j = V_{1j}X_1 + V_{2j}X_2 + \dots + V_{mj}X_m$$

Para todos los datos:

$$\begin{bmatrix} y_{1j} \\ \vdots \\ y_{nj} \end{bmatrix} = \begin{bmatrix} x_{11} & \dots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \dots & x_{nm} \end{bmatrix} \begin{bmatrix} v_{1j} \\ \vdots \\ v_{mj} \end{bmatrix}$$

En notación matricial:

$$\mathbf{y}_j = \mathbf{X}\mathbf{v}_j$$

Ahora este nuevo conjunto, \mathbf{y}_j será la componente principal j .

Para obtenerlo se debe encontrar el vector \mathbf{v}_j . El procedimiento para esto comienza encontrando la varianza máxima de la componente j , como se muestra a continuación:

$$Var(Y_j) = \frac{1}{n} \mathbf{y}'_j \mathbf{y}_j = \frac{1}{n} \mathbf{v}'_j \mathbf{X}' \mathbf{X} \mathbf{v}_j = \mathbf{v}'_j \mathbf{S} \mathbf{v}_j \quad (16)$$

Donde \mathbf{S} es la matriz de covarianzas.

Para esta ecuación existen infinitas soluciones en las mismas direcciones del espacio. Para que la maximización de la varianza de Y_j tenga solución se debe

imponer: $v_j'v_j = 1$, es decir, se toma un vector unitario en la dirección de la componente Y_j .

Ahora se tiene un problema de optimización en donde se busca maximizar la varianza de Y_j , resolviendo esto se obtendrían las componentes principales del conjunto inicial de datos.

2.3. Validación cruzada [11]

Es una técnica de validación altamente utilizada en proyectos de inteligencia artificial para realizar el entrenamiento y validación de un modelo. Esta técnica es de gran utilidad cuando se tiene una base de datos con pocas muestras.

Para implementarla se toman todos los datos y se dividen en N subconjuntos o Folds, luego se toman $N-1$ Folds como datos de entrenamiento y el Fold restante como datos de prueba. Este proceso se realiza N veces tomando como Fold de prueba un conjunto diferente en cada iteración, al final se tendrá un índice de desempeño por cada fold.

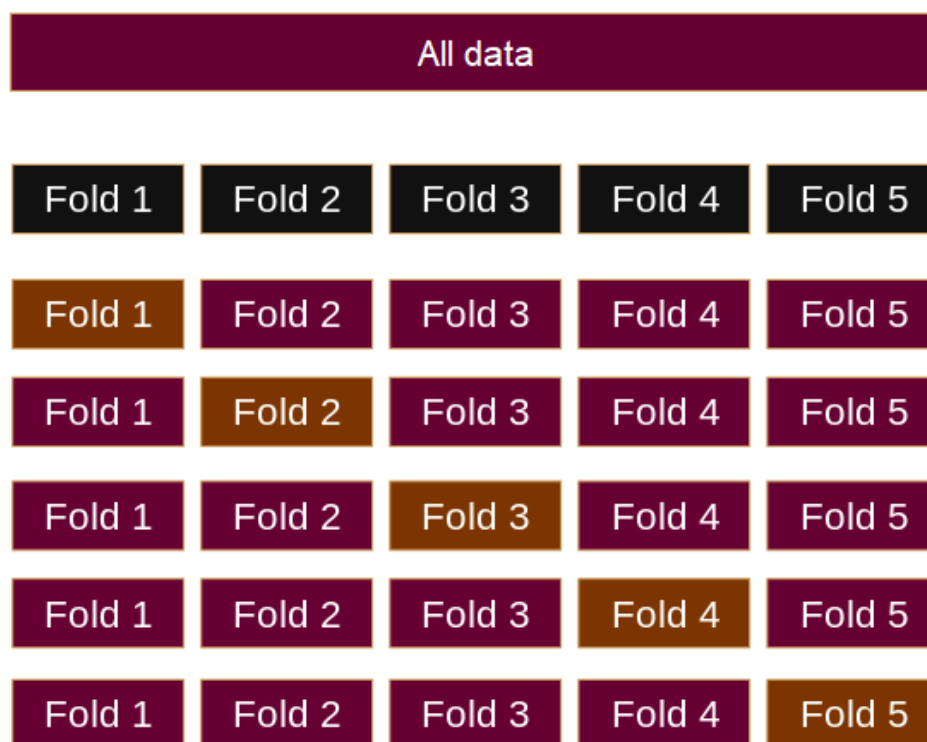


Figura 2. Validación cruzada con 5 folds

2.4. Clasificación

2.4.1. Máquinas de soporte vectorial (SVM) [4]

Una máquina de soporte vectorial o SVM (Máquinas de soporte vectorial) es una técnica de clasificación que se basa en la idea de minimización de riesgo estructural. Esta técnica se basa en encontrar un hiperplano de separación entre un conjunto de muestras X en un problema biclase maximizando el margen de separación entre los datos. El margen de separación se define como la distancia entre los vectores de soporte, los cuales están dados por las muestras más próximas al hiperplano de separación de cada clase.

Sea un conjunto de muestras $X = \{X_1, X_2, X_3, \dots, X_n\} \forall X_i \in R^n$, con un vector de etiquetas asociado $y = \{y_1, y_2, y_3, \dots, y_n\} \forall y_i \in R$ tal que cada y_i estará asociado a un X_i y este corresponderá a la clase de su muestra asociada. Para representar eso se muestra un vector \bar{U} que se refiere a una muestra cualquiera del conjunto, y se muestra el vector \bar{W} como el vector del hiperplano de separación.

La magnitud de la proyección del vector \bar{U} sobre el vector \bar{W} se expresa como $\bar{W} \cdot \bar{U}$. Si esta magnitud es mayor a un parámetro llamado c , la muestra representada por el vector \bar{U} es positiva, si es menor al parámetro c , la muestra representada por el vector \bar{U} es negativa. Se introduce un nuevo parámetro b definido como la distancia entre el hiperplano de separación y el vector de soporte. Con $c = -b$, la expresión para las muestras positivas sería $\bar{W} \cdot \bar{U} \geq c$, con el parámetro b la expresión para todas las muestras positivas sería $\bar{W} \cdot \bar{U} + b \geq 0$ y para las muestras negativas $\bar{W} \cdot \bar{U} + b \leq 0$. Así si la magnitud de la proyección del vector \bar{U} sobre el vector \bar{W} más el parámetro b es mayor a 0 la muestra será positiva, y si es menor a 0 la muestra será negativa.

La expresión para las muestras de los vectores de soporte sería, para las muestras positivas $\bar{W} \cdot \bar{X}_+ + b = 1$ y para las muestras negativas $\bar{W} \cdot \bar{X}_- + b = -1$, así para cualquier muestra positiva se cumple que $\bar{W} \cdot \bar{X}_+ + b \geq 1$ y para cualquier muestra negativa se cumple que $\bar{W} \cdot \bar{X}_- + b \leq -1$.

En la figura 3 se muestra una máquina de soporte vectorial con el vector \bar{U} de una muestra aleatoria y el vector \bar{W} del hiperplano de separación, y se muestran los vectores de soporte \bar{X}_+ y \bar{X}_-

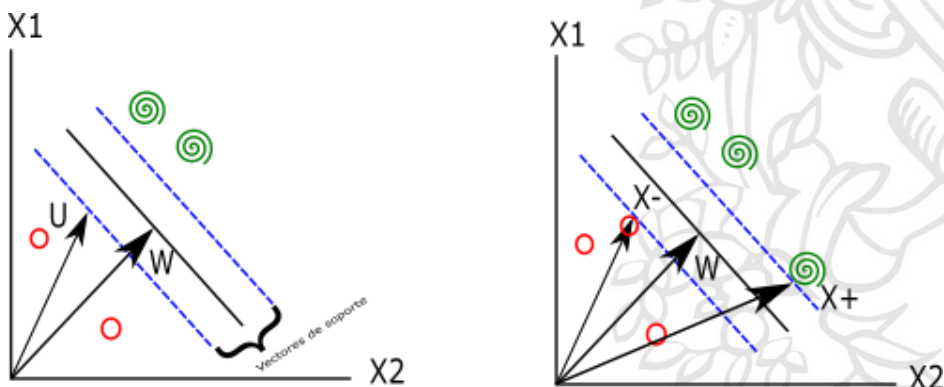


Figura 3. Máquina de soporte vectorial con el vector W del hiperplano y el vector U de una muestra al azar (Izquierda), máquina de soporte vectorial con el vector W y los vectores de soporte del hiperplano (Derecha).

$$\text{Sea } \overline{w} \cdot \overline{x}_i = x_i^w$$

Se define Y_i como +1 para todas las muestras positivas y como -1 para todas las muestras negativas, así la expresión general para todas las muestras $Y_i (X_i^w + b) - 1 \geq 0$ y para las muestras en los vectores de soporte $Y_i (X_i^w + b) - 1 = 0$, despejando se obtiene $X_+^w = 1 - b$ y $X_-^w = 1 + b$

Así la magnitud del margen de separación está dada como la resta de los vectores de soporte, esto se expresa de la siguiente forma $(X_+^w - X_-^w) \cdot \frac{\overline{w}}{\|\overline{w}\|}$. Reemplazando los valores de $X_+^w = 1 - b$ y $X_-^w = 1 + b$, el margen de separación estaría determinado por $(1 - b - 1 - b) \cdot \frac{\overline{w}}{\|\overline{w}\|}$, se muestra gráficamente en la figura 4.

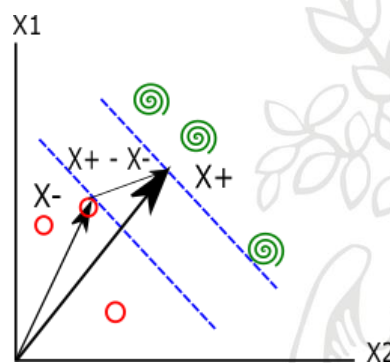


Figura 4. Vectores de soporte y su resta.

Así el margen de separación sería determinado por $\frac{2}{\|\overline{w}\|}$, el problema de optimización se basa es maximizar este margen, lo que se convierte en minimizar $\|\overline{w}\|$.

El desarrollo realizado es para máquinas de soporte vectorial con margen duro, es decir, no permite errores en la separación de los datos. Esto es útil en una situación ideal en donde los datos sean linealmente separables. En un caso real los datos son difícilmente linealmente separables, por lo que se introduce el concepto de margen suave. Este tipo de margen acepta la posibilidad de algunos errores y los penaliza según un nuevo parámetro llamado C. Así, aunque los datos no sean linealmente separables a totalidad se podrá encontrar un hiperplano de separación que ofrezca buenos resultados.

En el caso en que los datos no sean linealmente separables se dispone de funciones kernel, las cuales se encargan de hacer una transformación de los datos llevándolos a un nuevo hiperespacio en donde se puedan hacer realmente separables (Figura 5).

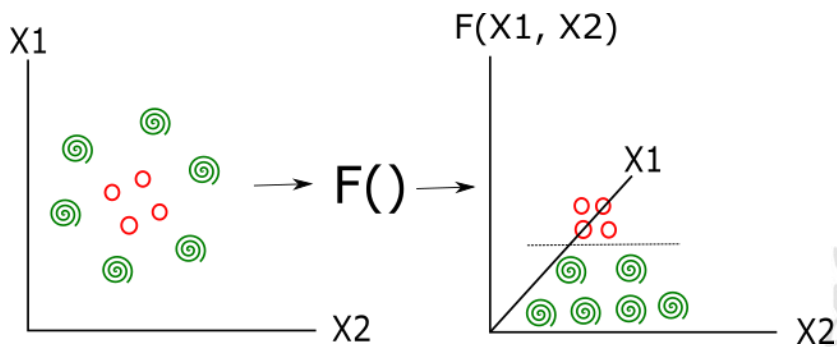


Figura 5. Espacio de características original (Izquierda), nuevo espacio de características con la función kernel (Derecha).

2.4.2. Bosques aleatorios (RF) [10]

Bosques aleatorios es un algoritmo que se basa en la teoría de los árboles de decisión para generar modelos de clasificación o regresión, mitigando el problema de alta varianza propio de los árboles de decisión.

Se supone un dataset de características X , con su respectivo vector de etiquetas Y .

El algoritmo de bosques aleatorios dividirá la base de datos en m partes iguales, donde m es el número de estimadores, es decir, el número de árboles dentro del bosque de decisión. Cada árbol se generará con una profundidad diferente. Así se tendrán diversos árboles generados con una parte diferente del conjunto de datos.

Con la llegada de una nueva muestra cada árbol de decisión generará una etiqueta asociada. La etiqueta final se obtiene como la moda de las etiquetas generadas, es decir, voto por mayoría. El proceso se muestra gráficamente en la figura 6.

Un conocido problema de los árboles de decisión es su alta varianza, dicho problema es mitigado al general un bosque de varios árboles, ya que la varianza final será menor a la varianza de cada árbol.

Sean m árboles la varianza final será:

$$\text{Var} \left[\frac{1}{m} \sum_{i=0}^m Y_i \right] = \frac{1}{m^2} \sum_{i=0}^m \text{Var}(Y_i) = \frac{1}{m^2} m \sigma^2 = \frac{\sigma^2}{m} \quad (17)$$

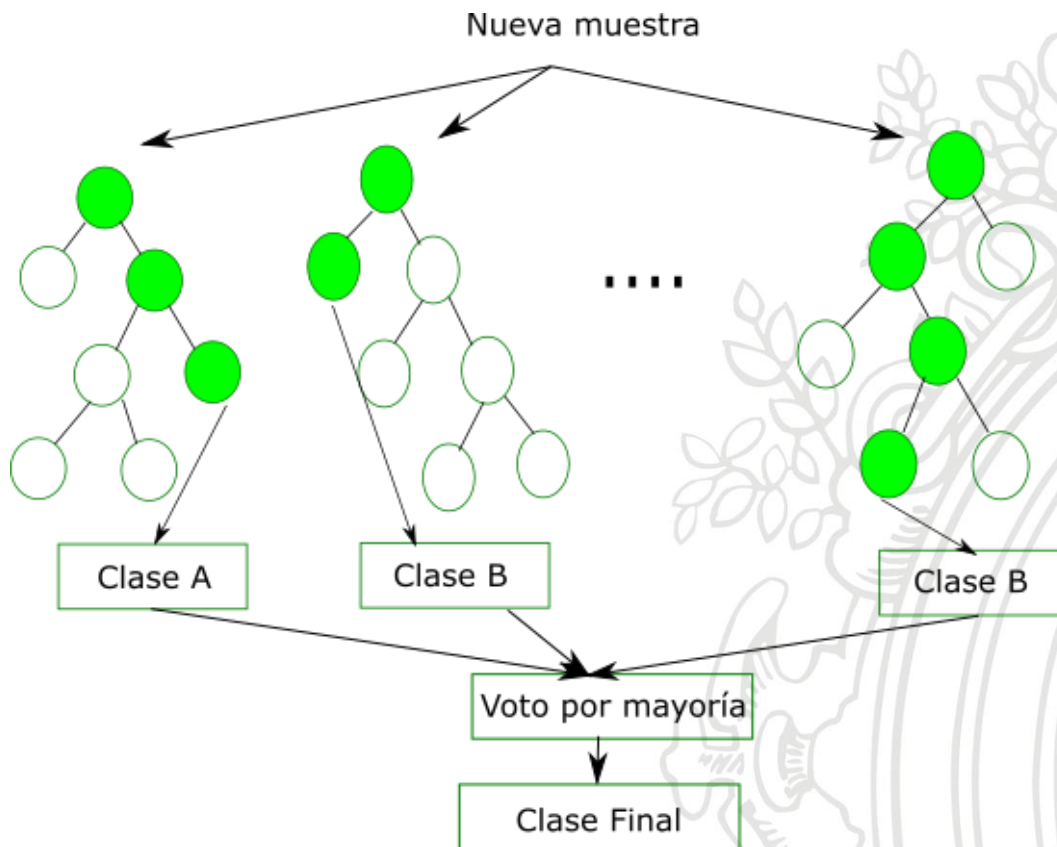


Figura 6. Bosques aleatorios

2.4.3. Curva ROC [8]

La curva (Receiver Operating Characteristic) es una herramienta que compara 1-especificidad contra la sensibilidad para cada valor del umbral o punto de corte en la escala de resultados del experimento. En un modelo perfecto la curva ROC tendrá sensibilidad 1 para todo punto de corte y la especificidad cambiará según el punto de corte, es decir, las 2 clases tienen una separación perfecta, esto genera resultado una gráfica tipo escalón (Figura 7).

Es de gran utilidad en la evaluación de clasificadores ya que proporciona información sobre la exactitud del modelo y la separación entre las clases.

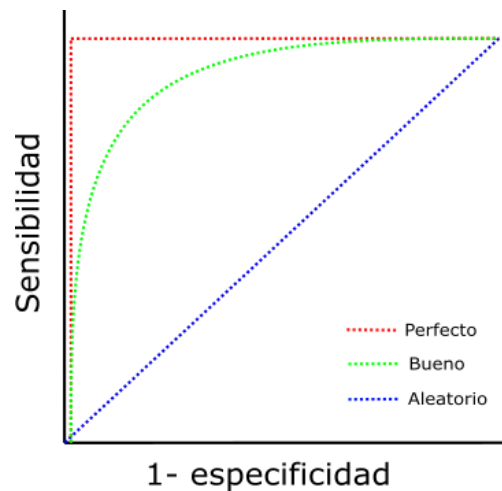


Figura 7. Curvas ROC para diferentes tipos de desempeño

Si la curva ROC es empinada mejor será el desempeño del modelo representado, ya que el área bajo la curva representa la efectividad del sistema. Entonces si la curva ROC es cercana a la curva azul observada en la figura 7, el sistema no estará prediciendo, sino que sus resultados serán aleatorios.

2.4.4. Matriz de confusión [4]

La matriz de confusión es una técnica muy usada para evaluar el desempeño de un clasificador. Esta se encarga de comparar los resultados de un clasificador con los valores reales mediante diferentes coeficientes. En un problema biclase esta matriz es de tamaño 2x2, es decir, tiene 4 coeficientes los cuales son, verdaderos positivos (clase 1 bien clasificados), falsos negativos (clase 1 mal clasificados), falsos positivos (clase 2 mal clasificados) y verdaderos negativos (clase 2 bien clasificados). Se muestra la matriz de confusión en la Figura 8.

		Predicho	
		Positivos	Negativos
Real	Positivos	Verdaderos Positivos (VP)	Falsos Negativos (FN)
	Negativos	Falsos Positivos (FP)	Verdaderos Negativos (VN)

Figura 8. Matriz de confusión

2.5. Regresión

2.5.1. Regresión por vectores de soporte (SVR) [5]

Debido al éxito de las SVM surge otra técnica que parte de la misma base, pero con el fin de encontrar una función de regresión que se ajuste a los datos. Una regresión por vectores de soporte SVR (Regresión por vectores de soporte) es una técnica descendiente de las SVM, la cual generaliza el proceso de optimización, generando una función continua e introduciendo el concepto de la región insensitiva ε . Este tubo reformula el problema de optimización en encontrar un hiper-tubo que mejor aproxime la función (Figura 9).

Similar a las SVM con el margen de separación h generado por los vectores de soporte, las SVR introducen su margen por medio del parámetro ε que generará la región insensitiva ε , los datos que estén dentro de esta zona son tolerados, es decir, no tienen un error asociado. Esto se hace con el fin de minimizar el error del regresor. Cuando el problema de regresión no es lineal, se usa la función kernel para mapear los datos a un espacio de mayor dimensionalidad y resolver allí el problema. Además, se incluye el parámetro C propio de las SVM de margen suave, el cual penaliza los errores que se presenten.

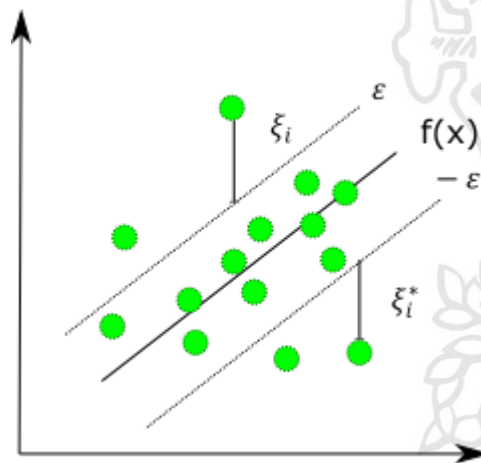


Figura 9. Regresión por vectores de soporte

El problema de optimización para resolver de la SVR es:

$$\min \left(\frac{1}{2} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \right) \quad (18)$$

Donde $\|w\|^2$ es la norma del vector w perpendicular al hiperplano, C es el parámetro de penalización del error y ξ_i y ξ_i^* son los errores asociados a los datos que se encuentren sobre y debajo del hiper-tubo ε .

2.5.2. Regresión por bosques aleatorios (RF para regresión) [10]

El algoritmo de Regresión por bosques aleatorios es igual al algoritmo usado para clasificación, pero en el último paso no se usa la moda de las predicciones de los diferentes árboles, sino que se usa el promedio de dichas decisiones, generando así una salida continua y no discreta.

2.5.3. Coeficiente de correlación de rangos de Spearman [6]

El coeficiente de correlación de Spearman mide la relación monótona entre 2 variables continuas u ordinales. En una relación monótona entre 2 variables, estas cambian al mismo tiempo, pero no necesariamente a un ritmo constante.

2.5.4. Error medio Absoluto (MAE)

Este índice determina la media en la diferencia absoluta de los errores. Con este valor se obtiene una puntuación lineal, es decir, se ponderan por igual todas las diferencias individuales.

$$MAE = \frac{1}{N} \sum_{i=0}^{N-1} |y_i - \hat{y}_i| \quad (19)$$

3. Metodología

3.1. Base de datos [12]

La base de datos a utilizar consiste es una serie de videos captado con el sensor de profundidad Kinect. En los videos se encuentra un paciente diagnosticado con la enfermedad de Parkinson realizando el movimiento toe-heel tapping, o toque punta-talón.

En la figura 10 se muestra un frame de los videos captados por el sensor Kinect



Figura 10. Frame de un video de la base de datos

3.2. Seguimiento del talón

En esta etapa se realizó de manera manual el seguimiento del movimiento del talón del paciente mientras realizaba la actividad de toque punta-talón con el pie izquierdo y luego con el pie derecho. Así se capturó la posición y el tiempo en el que el talón estaba a su máxima altura y el tiempo en que tocaba el suelo. Para esto se usó el programa Tracker el cual permite realizar un seguimiento frame por frame y un etiquetado de la posición del objeto de interés, así se obtuvo una tabla con los valores de la posición en Y del talón etiquetado y su respectivo tiempo generando como resultado una señal triangular con el tiempo en el eje X y la posición Y del talón en el eje Y. También se estudió la aplicación del algoritmo de flujo óptico para realizar dicho seguimiento de forma automática.

3.3. Extracción de características

Luego de obtener la señal de movimiento en la etapa anterior, es necesario obtener las diferentes características para generar la Evaluación. Con la señal se usó del lenguaje de programación Python se obtuvo de la señal el jitter absoluto, el jitter relativo, la desviación estándar del jitter, el shimmer absoluto, el shimmer relativo, el shimmer dB, la desviación estándar del shimmer, el promedio de la energía, y la desviación estándar de la energía. Estas características se obtuvieron para cada pie, así se obtuvo en total 18 características por cada paciente.

Luego se aplicó la técnica de análisis de componentes principales (PCA) con el fin de reducir la dimensionalidad de las características conservando más del 90% de la varianza de los datos.

3.4. Evaluación

El desarrollo de esta etapa se realizó en 2 pasos, afinación de parámetros y generación del modelo final.

En la afinación de parámetros se entrenaron modelos de clasificación y regresión con el fin de predecir el índice UPDRS-Lower-Limbs, para esto se dividió la base de datos en conjuntos de entrenamiento y test pero debido a la baja cantidad de muestra, se usó la técnica de Validación cruzada con 5 folds para entrenar los modelos. En cada iteración del algoritmo de Validación cruzada se tomó el conjunto de entrenamiento y se hizo una afinación de parámetros por medio de una malla de búsqueda la cual hace un barrido por un rango de valores en los diferentes parámetros y usando Validación cruzada con 4 folds la malla de búsqueda entrega el mejor conjunto de parámetros encontrado con los 4 folds. Se obtuvo resultados del conjunto de test probado en el modelo con el mejor conjunto de parámetros de la malla de búsqueda

y por último se obtiene la media y la desviación estándar de la malla de búsqueda del índice de desempeño del modelo a validar. Después de esto se tiene 1 conjunto de parámetros por cada fold, en total se tienen 5 conjuntos de parámetros afinados. Para los parámetros finales se obtuvo la moda de los 5 conjuntos generando un conjunto de parámetros final.

Se muestra la división de la base de datos en la figura 11:

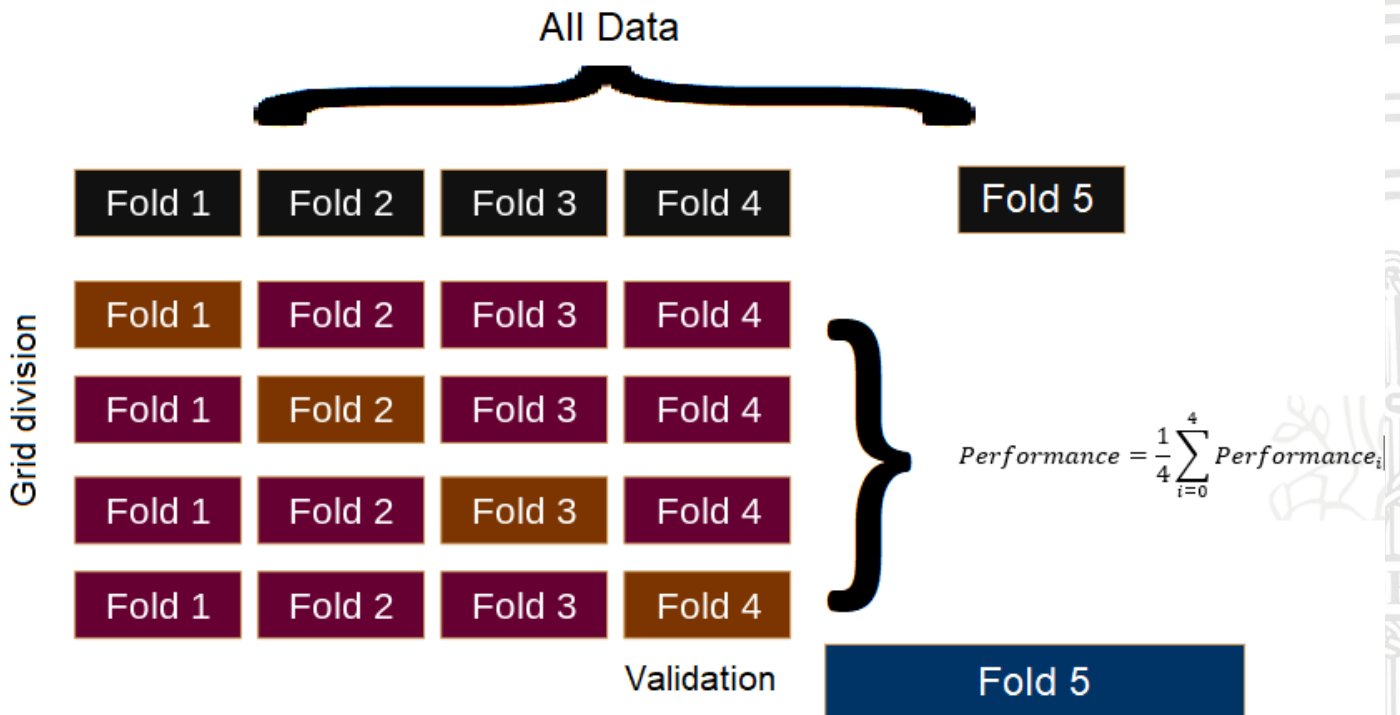


Figura 11. División de la base de datos para el algoritmo

En la generación y evaluación del modelo final se entrenó con los parámetros finales, se divide la base de datos en N_folds y se usa validación cruzada para evaluar el modelo. En cada iteración de la validación cruzada se guardan los valores de las etiquetas reales y predichas del fold de validación, concatenándolos en un solo vector cada uno. Al final de la validación se obtienen los índices y gráficas de los vectores concatenados de las etiquetas reales y predichas. Este proceso se realizó con las características normales y con las componentes obtenidas con PCA.

A continuación, en la figura 12, se muestra el diagrama de flujo del algoritmo descrito.

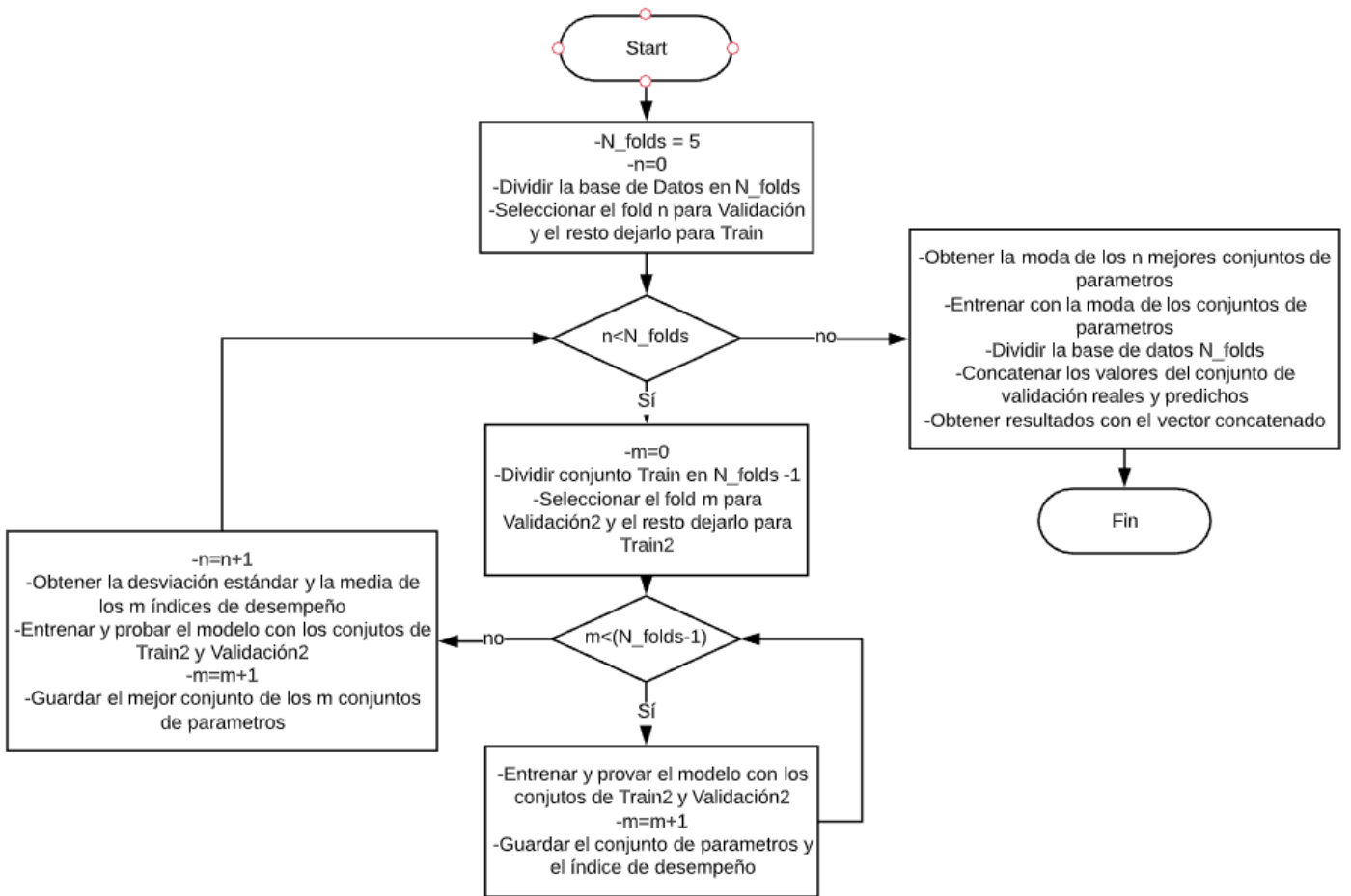


Figura 12. Algoritmo de generación y evaluación de los modelos de evaluación

Con el algoritmo general definido se procedió a implementarlo con las diferentes técnicas e índices de desempeño.

3.4.1. Clasificación

Para la clasificación se debe dividir el vector de etiquetas en 2 clases, esto se hizo tomando como un umbral la mediana de los índices, así se etiquetó como 0 los que tuviesen un índice menor a dicho umbral y como 1 aquellos que tuviesen un índice menor a este umbral.

Se dividió el índice UPDRS en 2 clase, para esto se encontró la mediana de los índices la cual fue 29.5 y aquellos que tuviesen una puntuación UPDRS menor a esto se etiquetó como 0 y a aquellos que tuviesen un índice mayor se etiquetó como 1. A continuación, en la figura 13, se muestra el histograma de la distribución de los datos y la división por la mediana.

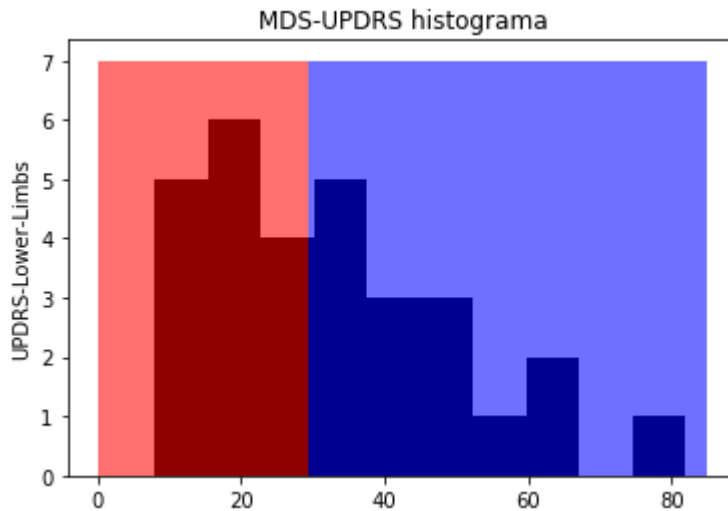


Figura 13. Histograma de los datos y división por la mediana, rojo severidad baja, azul severidad alta.

Como clasificación se utilizaron las siguientes técnicas:

- SVM con Kernel Lineal, cuyo parámetro para afinar es C .
- SVM con Kernel Gaussiano, cuyos parámetros para afinar son C , y γ .
- Bosques aleatorios, cuyos parámetros para afinar son: Número de estimadores, la máxima profundidad, Número mínimo de muestras para división.

Para estos el índice de desempeño usado fue la tasa de aciertos.

3.4.2. Regresión

Para la regresión se usó el índice UPDRS-Lower-Limbs, el cuál presenta un rango para los pacientes evaluados de 8-82.

Como regresión se usaron:

- SVR con Kernel Lineal, cuyos parámetros para afinar son ϵ y C .
- SVR con Kernel Gaussiano, cuyos parámetros para afinar son ϵ , C , y γ .
- Bosques aleatorios, cuyos parámetros para afinar son: Número de estimadores, la máxima profundidad, Número mínimo de muestras para división.

Para estos el índice de desempeño fue el error medio absoluto (MAE) y el coeficiente de correlación de rangos de Spearman.

Los rangos de valores probados para cada uno de los índices fueron:
 gamma: [1e3,1e2,1,1e-1,1e-2,1e-3, 1e-4,1e-5,1e-6,1e-7,1e-8]
 C: [1e-4,1e-3,1e-2,1e-1,1,1e2,1e3]
 epsilon: [1e3,1e2,1,1e-1,1e-2,1e-3, 1e-4,1e-5,1e-6,1e-7,1e-8]
 Número de estimadores: [4, 8, 16]
 máxima profundidad: [4, 8, 16]
 Número mínimo de muestras para división: [2, 8, 16]

Se resume la información anterior en la tabla 1:

Técnica		Parámetros para afinar	Índice de desempeño
SVM	Kernel Lineal	C	Tasa de aciertos
	Kernel Gaussiano	C, gamma	
Bosques aleatorios		Número de estimadores, la máxima profundidad, Número mínimo de muestras para división.	
SVR	Kernel Lineal	Epsilon, C	Índice de correlación de rangos de Spearman y MAE
	Kernel Gaussiano	Epsilon, C, gamma	
Random-Forest-Regressor		Número de estimadores, máxima profundidad, Número mínimo de muestras para división.	

Tabla 1. Descripción de técnicas usadas con sus respectivos parámetros e índices.

4. Resultados y análisis

4.1. Seguimiento del balón

4.1.1. Manual

Mediante Tracker se obtuvo manualmente la señal de movimiento del balón de cada video.

En la figura 14 se muestra el programa Tracker y una señal obtenida por el seguimiento manual

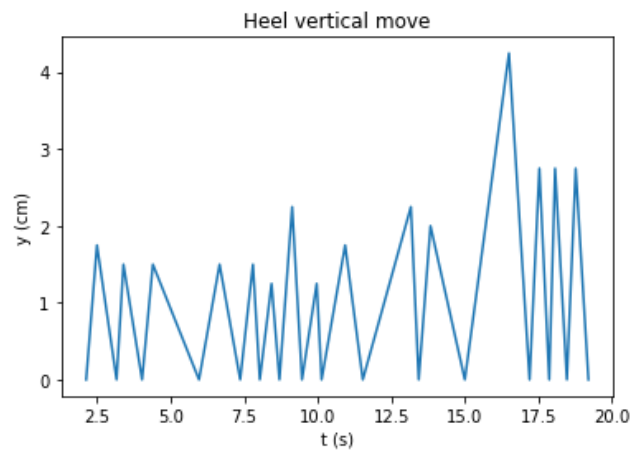
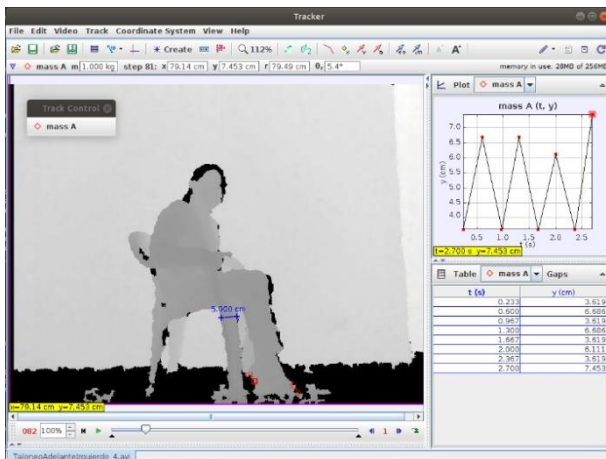


Figura 14. A la derecha el programa tracker, a la izquierda los datos de movimiento vertical

4.1.2. Automática

De forma automática se utilizó el algoritmo de Flujo óptico para realizar el seguimiento de forma automática. Con el algoritmo se obtuvieron 2% de videos en donde se detectó y se siguió bien el balón, 42% en donde se detectó el balón, pero no se realizó un buen seguimiento y 56% de videos en donde no se detectó el balón.

En la figura 15 se muestra el seguimiento realizado de manera automática.

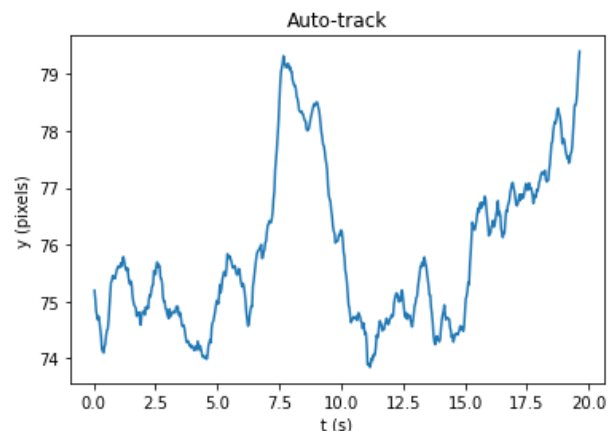


Figura 15. A la derecha detección y seguimiento por flujo óptico, a la izquierda gráfica obtenida por el seguimiento automático.

Los errores asociados al seguimiento automático son debidos a que es necesario realizar una buena identificación del objeto antes de aplicar el algoritmo de flujo óptico, ya que es necesario saber a qué vecindad de píxeles se les aplicará el algoritmo. Para mejorar los errores se puede captar la imagen RGB y la información de la profundidad para aplicar un algoritmo más robusto de detección de zona de interés.

El algoritmo de flujo óptico tuvo un rendimiento bajo debido a la mala identificación previa del talón, ya que el algoritmo se encarga de realizar el seguimiento de un objeto previamente identificado. Si el objeto es bien identificado previamente es posible que el seguimiento se pierda en algún punto, es por esto que el porcentaje de seguimiento exitoso es tan bajo.

4.2. Extracción de características

Debido a la alta tasa de error del algoritmo aplicado se decidió usar las señales extraídas de forma manual por medio del software Tracker.

A continuación, en la figura 16, se muestra el boxplot de la Energía, el Jitter absoluto y el Shimmer absoluto del pie izquierdo de los pacientes, con el índice UPDRS-Lower-Limbs ya dividido en las clases con severidad media-baja y media-alta.

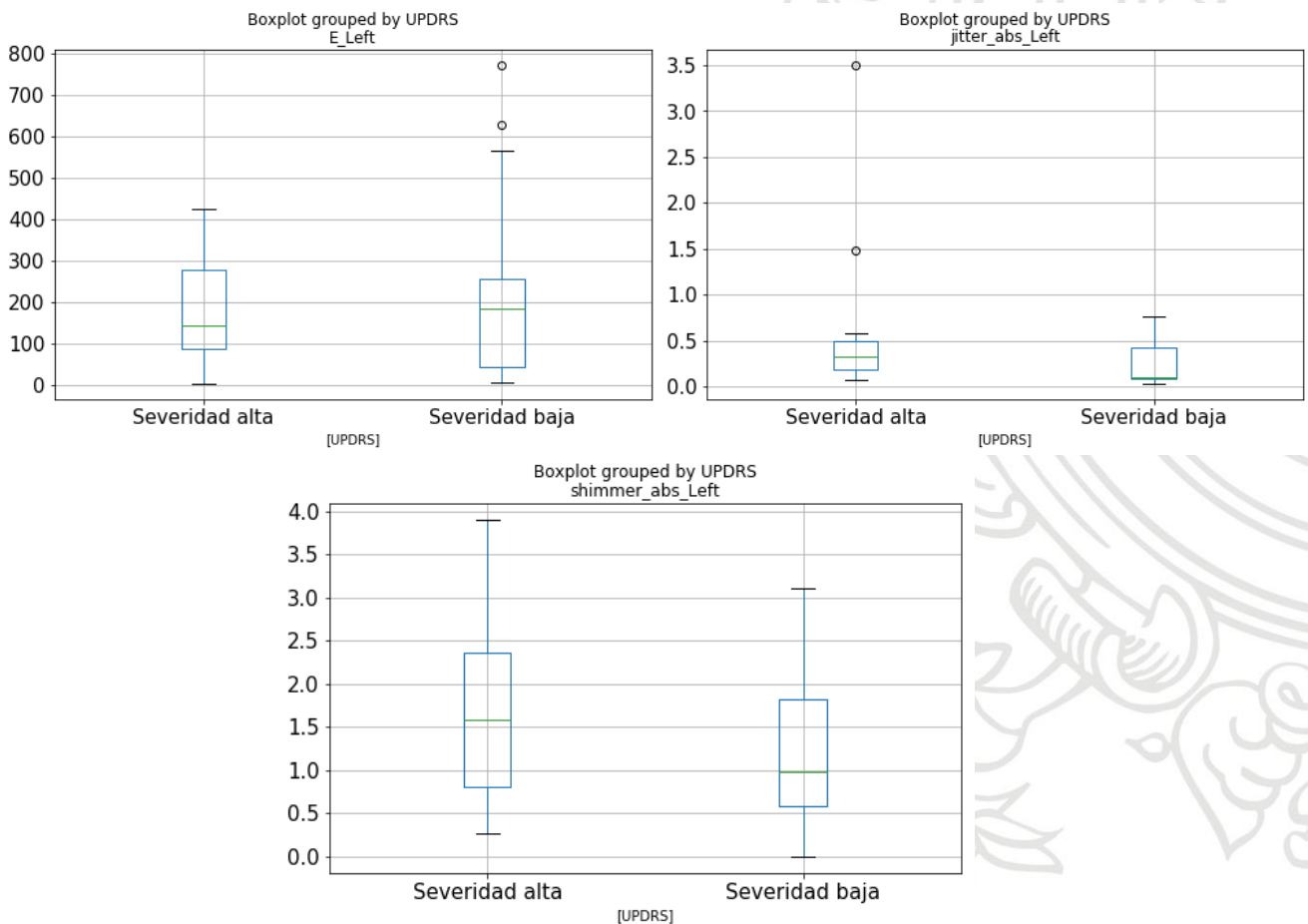


Figura 16. Boxplot de la energía, el jitter absoluto y el shimmer absoluto del pie izquierdo

Se evidencia que las características son discriminantes según las clases debido a que existe una separación entre cajas, lo que da indicios de que el problema puede tener resultados aceptables en los índices de desempeño. Se realiza una prueba T de Student para verificar la diferencia entre clases en todas las características. Los resultados se muestran en la tabla 2.

Característica	T_value	P_value
jitter_std_Left	-1.22	0.23
jitter_abs_Left	-1.38	0.18
jitter_rel_Left	-1.79	0.08
shimmer_std_Left	-0.37	0.71
shimmer_abs_Left	-1.06	0.29
shimmer_rel_Left	-1.90	0.07
shimmer_dB_Left	-2	0.05
E_std_Left	-0.87	0.39
E_Left	0.62	0.54
jitter_std_Right	-1.50	0.14
jitter_abs_Right	-1.69	0.10
jitter_rel_Right	-2.03	0.05
shimmer_std_Right	0.37	0.71
shimmer_abs_Right	0.12	0.91
shimmer_rel_Right	-0.82	0.42
shimmer_dB_Right	-0.88	0.38
E_std_Right	0.15	0.88
E_Right	0.86	0.39

Tabla 2. Prueba T de Student para las características.

Se observa que las características más discriminantes son el shimmer dB de la pierna izquierda y el jitter relativo de la pierna derecha, ya que tienen un valor T mayor o igual a 2 y un valor de p menor a 0.05 lo que indica confiabilidad en el resultado del valor T.

A todos los datos se les aplicó el algoritmo de PCA con varianza acumulada de mínimo 90%, esto redujo el espacio de 18 características a 2 componentes principales, generando la distribución de datos formada en la figura 17:

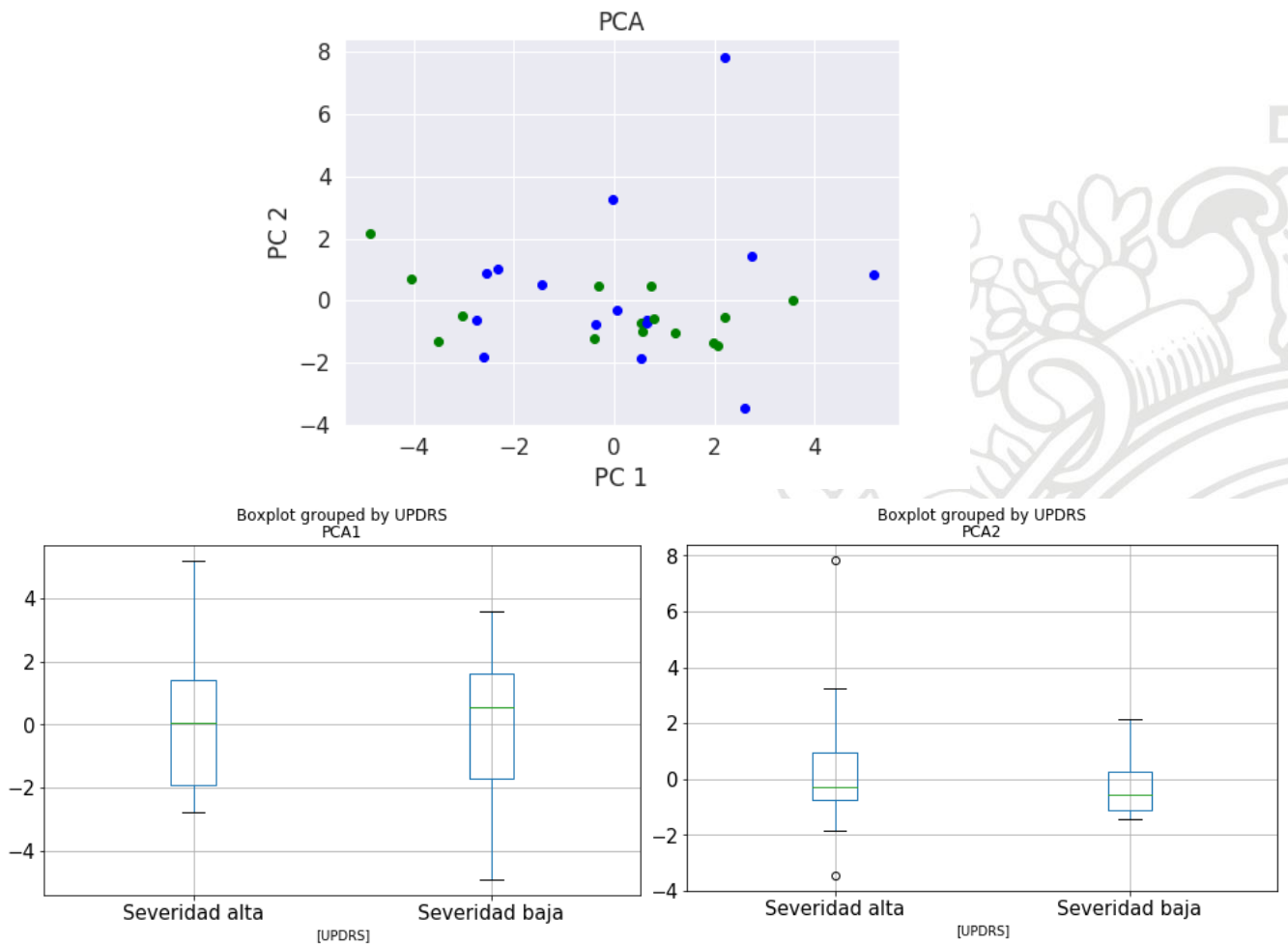


Figura 17. Grafica de las 2 componentes principales obtenida y boxplot de las 2 componentes.

Se puede notar que después de aplicado el algoritmo, las características no tienen mayor separación en el espacio de las componentes principales. Además, en el boxplot se muestra que si bien existe una separación entre las cajas hay muy pocas características, lo que podría resultar en un desempeño más bajo que con las 18 características completas.

4.3. Evaluación

En clasificación se trabajó con el índice UPDRS-Lower-Limbs ya dividido en las clases de baja y alta severidad.

4.3.1. SVM y RF

Luego de la afinación de parámetros y entrenamiento de la SVM, se usaron todos los datos para obtener la tasa de aciertos del modelo usando toda la base de datos.

A continuación, se muestran los parámetros afinados, la matriz de confusión, la curva ROC y la distribución del índice de decisión de la clasificación de toda la base de datos.

Para kernel lineal, el parámetro C es 100. Se obtuvo una tasa de aciertos de 60%.

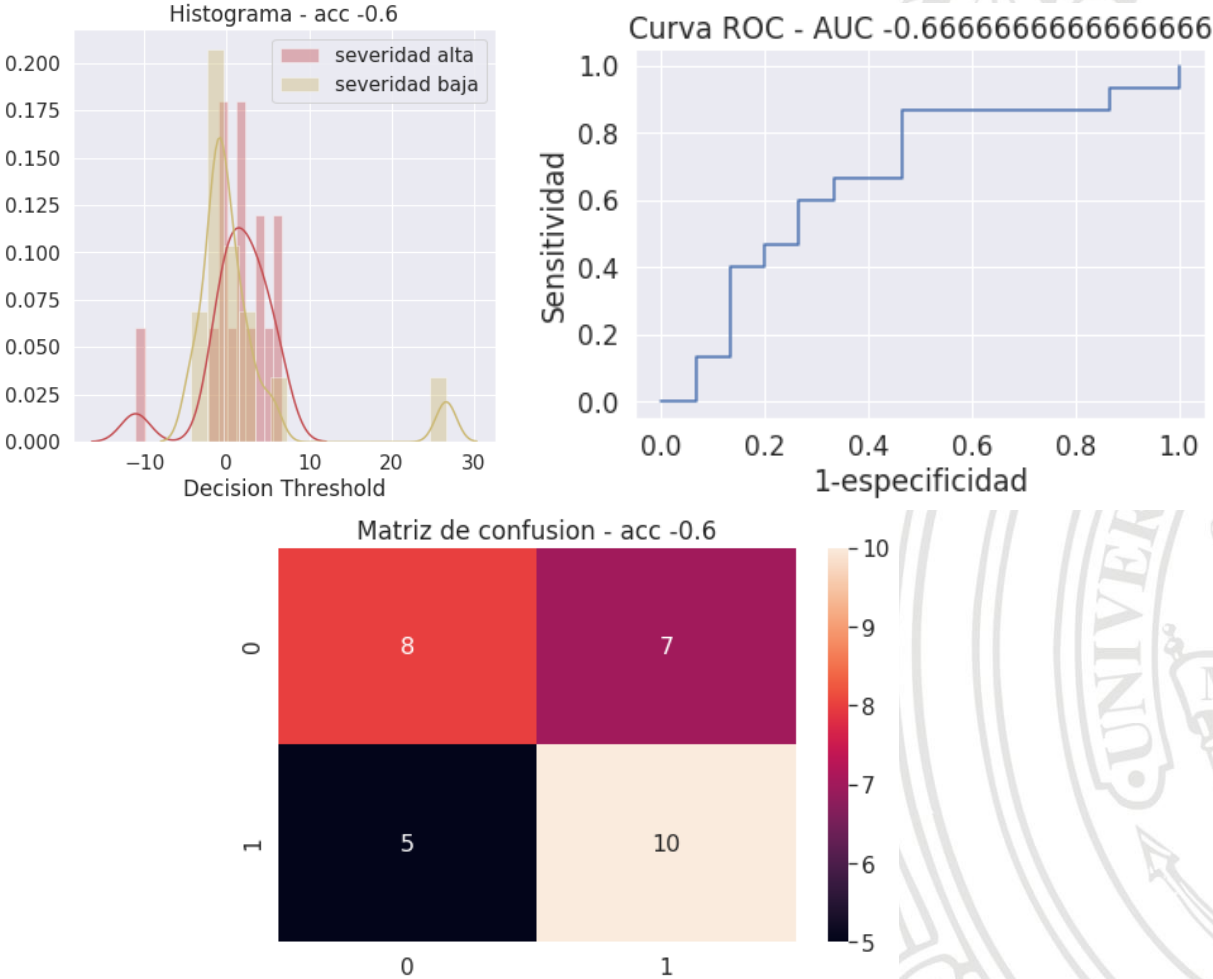


Figura 18. Histograma, curva ROC y matriz de confusión de SVM con kernel lineal

Los histogramas, la curva ROC y la matriz de confusión muestran que el sistema (Figura 18) produce una baja separación de los datos.

Con las componentes de PCA el parámetro C fue 1000 y se obtuvo una tasa de aciertos de 37%.

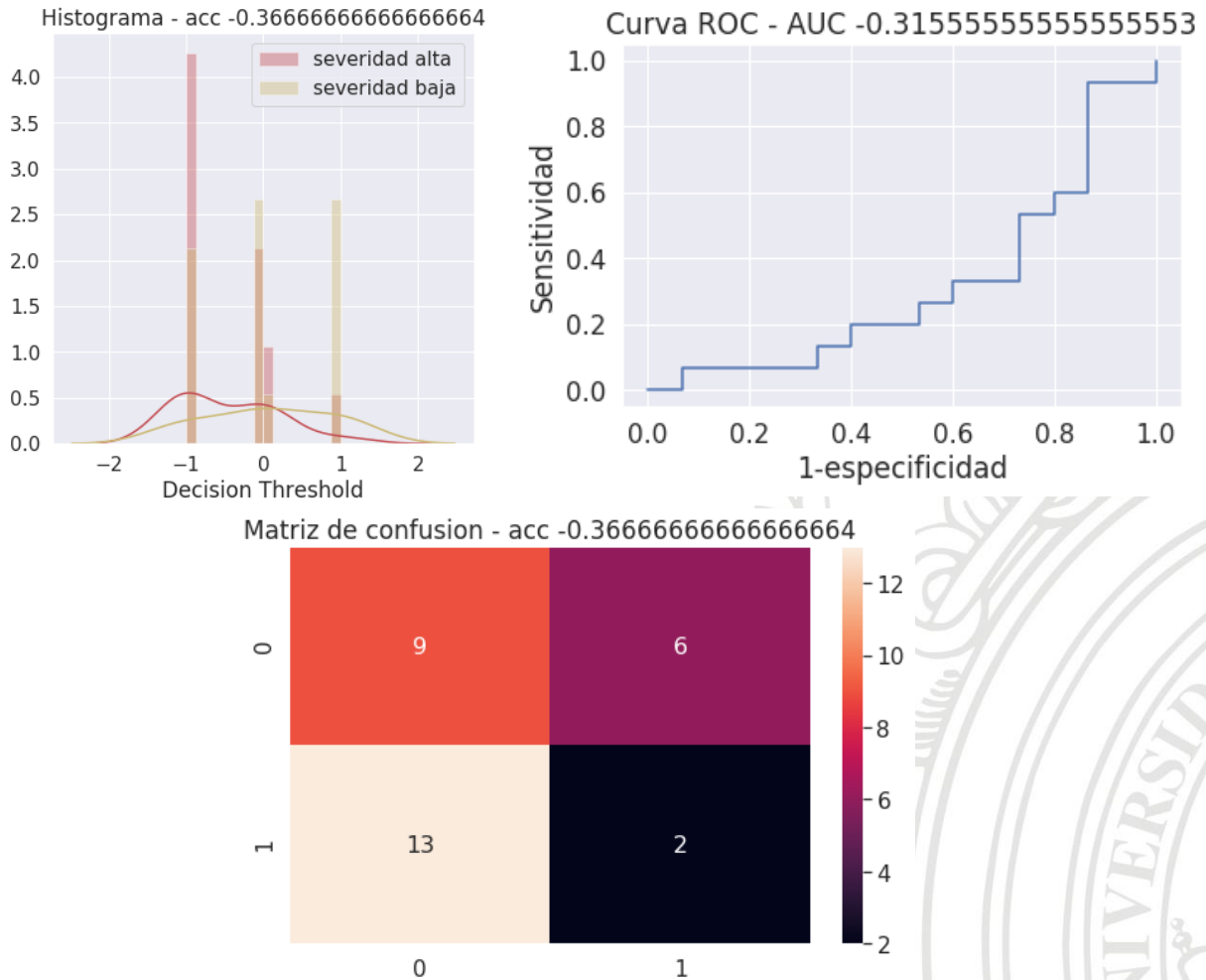


Figura 19. Histograma, curva ROC y matriz de confusión de SVM con kernel lineal y con las componentes principales

Se puede evidenciar en la figura 19 que con las componentes principales obtenidas por PCA el desempeño del sistema bajó sustancialmente comparado al sistema entrenado con todas las características.

Para kernel RBF (Radial Basis Function), el parámetro C es 100, y el parámetro gamma es 0.01. Se obtuvo una tasa de aciertos de 70% (Figura 20).

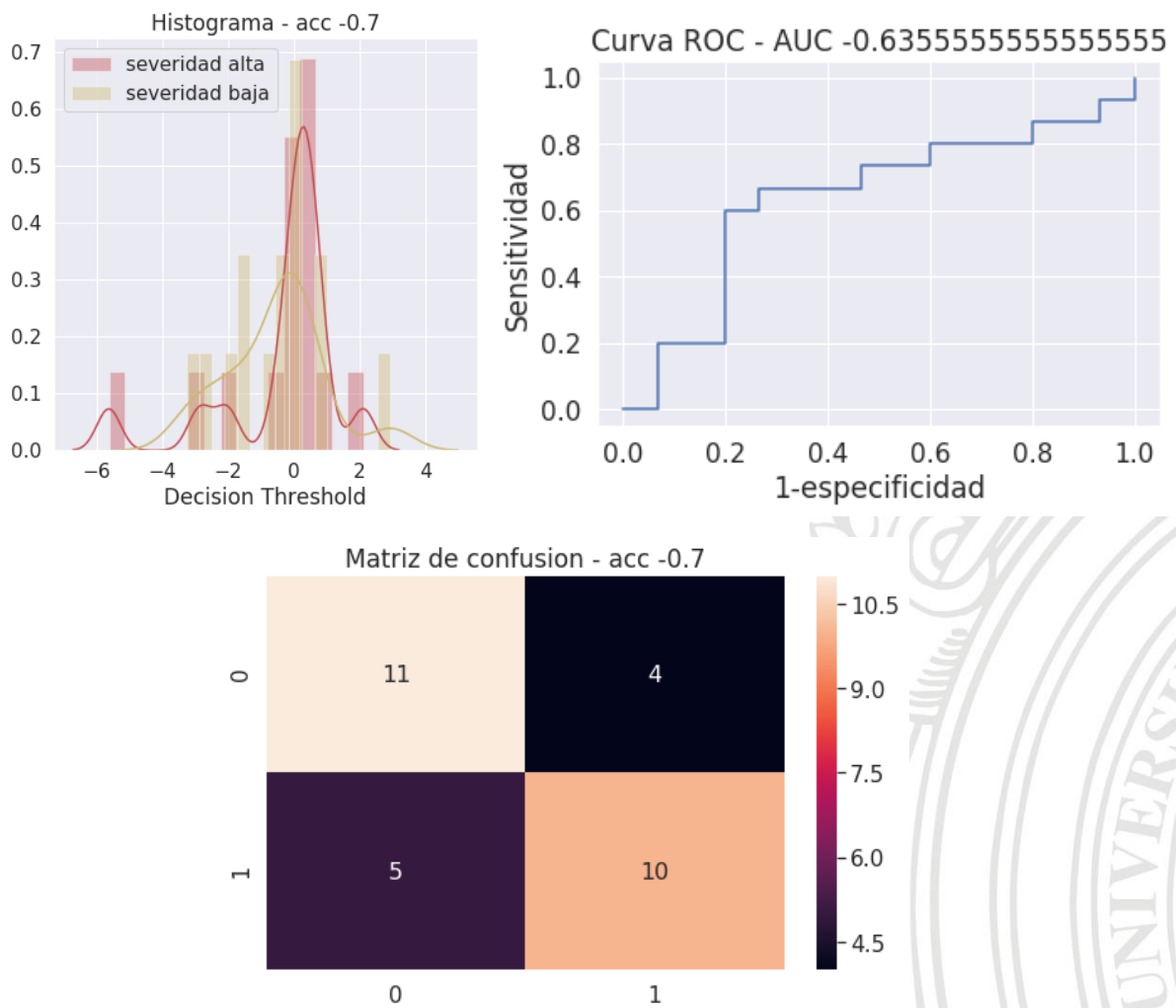


Figura 20. Histograma, curva ROC y matriz de confusión de SVM con kernel RBF

Con las componenetes de PCA, el parámetro C fue 1000 y gamma fue 0.01. Se obtuvo una tasa de aciertos de 40%.

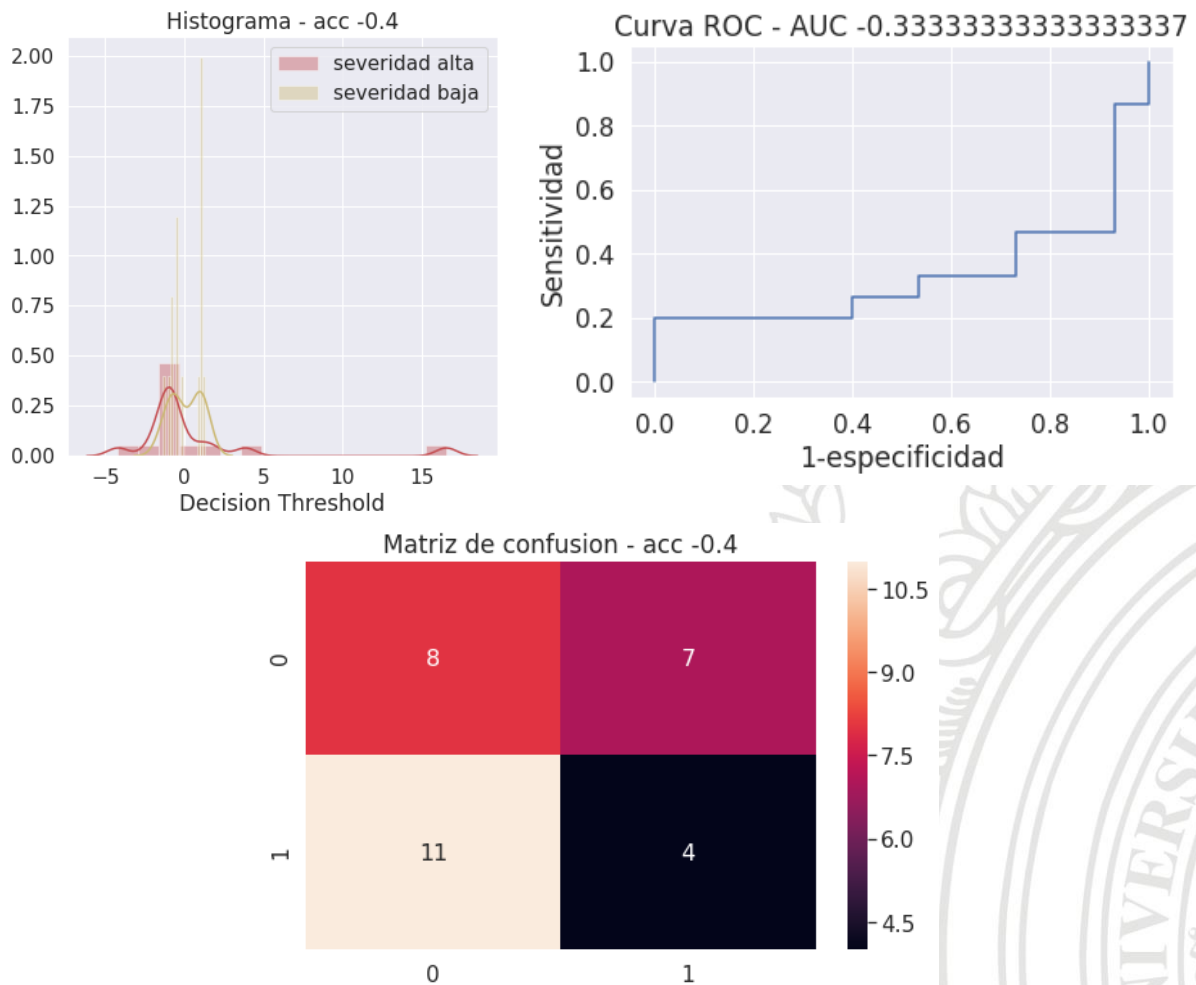


Figura 21. Histograma, curva ROC y matriz de confusión de SVM con kernel RBF y con las componentes principales

Al igual que el modelo de la SVM con kernel lineal se muestra una baja capacidad de generalización de los datos lo que indica un sub-entrenamiento (Figura 21).

Los parámetros afinados para RF fueron, número de estimadores igual a 16, la máxima profundidad fue 4 y el número mínimo de muestras para división fue 2. Se obtuvo una tasa de aciertos de 73%.

Como Bosques aleatorios no se basa en un umbral de decisión para la clasificación, pero se usa la probabilidad de pertenencia a la clase 1 para obtener el histograma. A continuación, en la figura 22, se muestra la curva ROC y la matriz de confusión para la clasificación

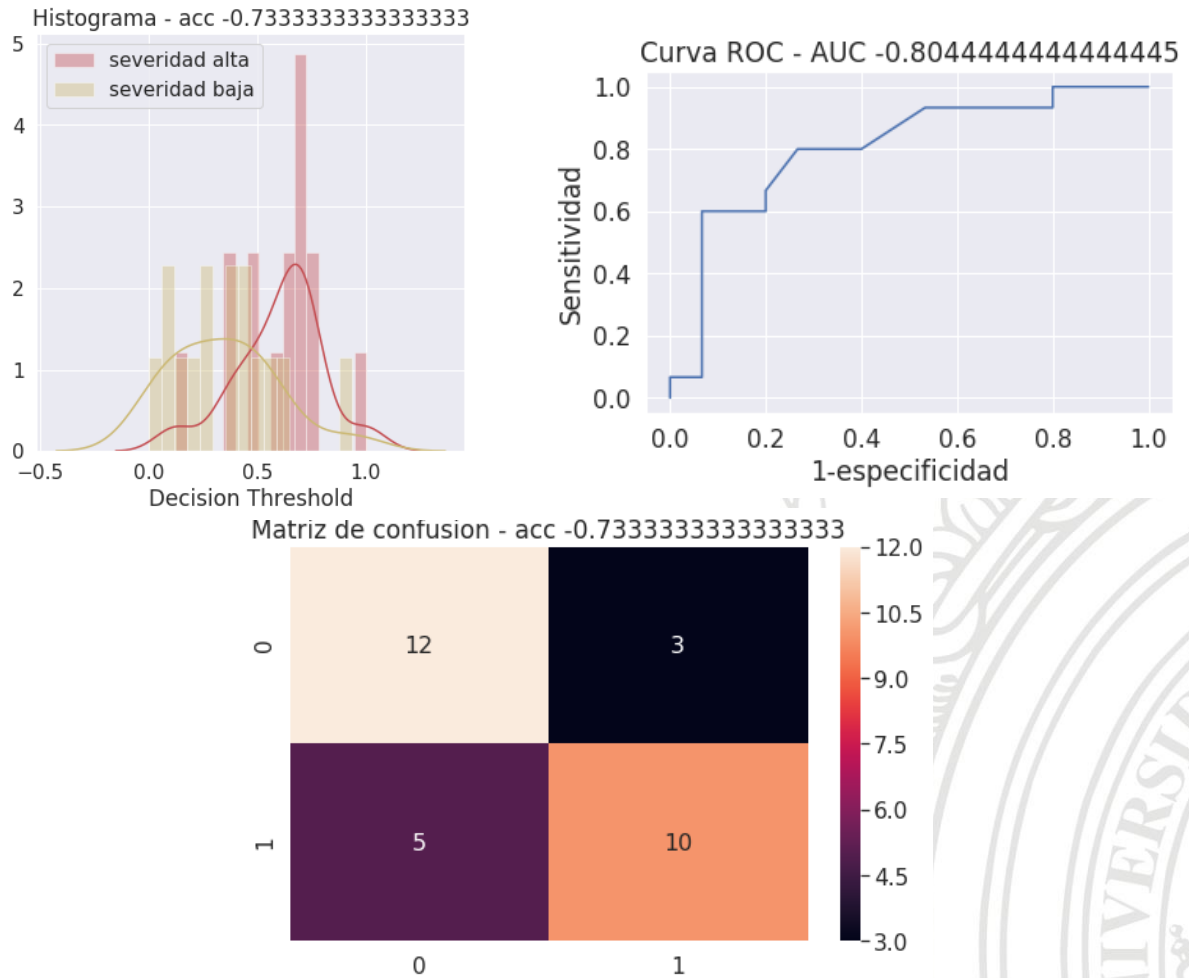


Figura 22. curva ROC y matriz de confusión de RF

Con las componenetes de PCA, el número de estimadores fue igual a 8, la máxima profundidad fue 4 y el número mínimo de muestras para división fue 2. Se obtuvo una tasa de aciertos de 66% (Figura 23).

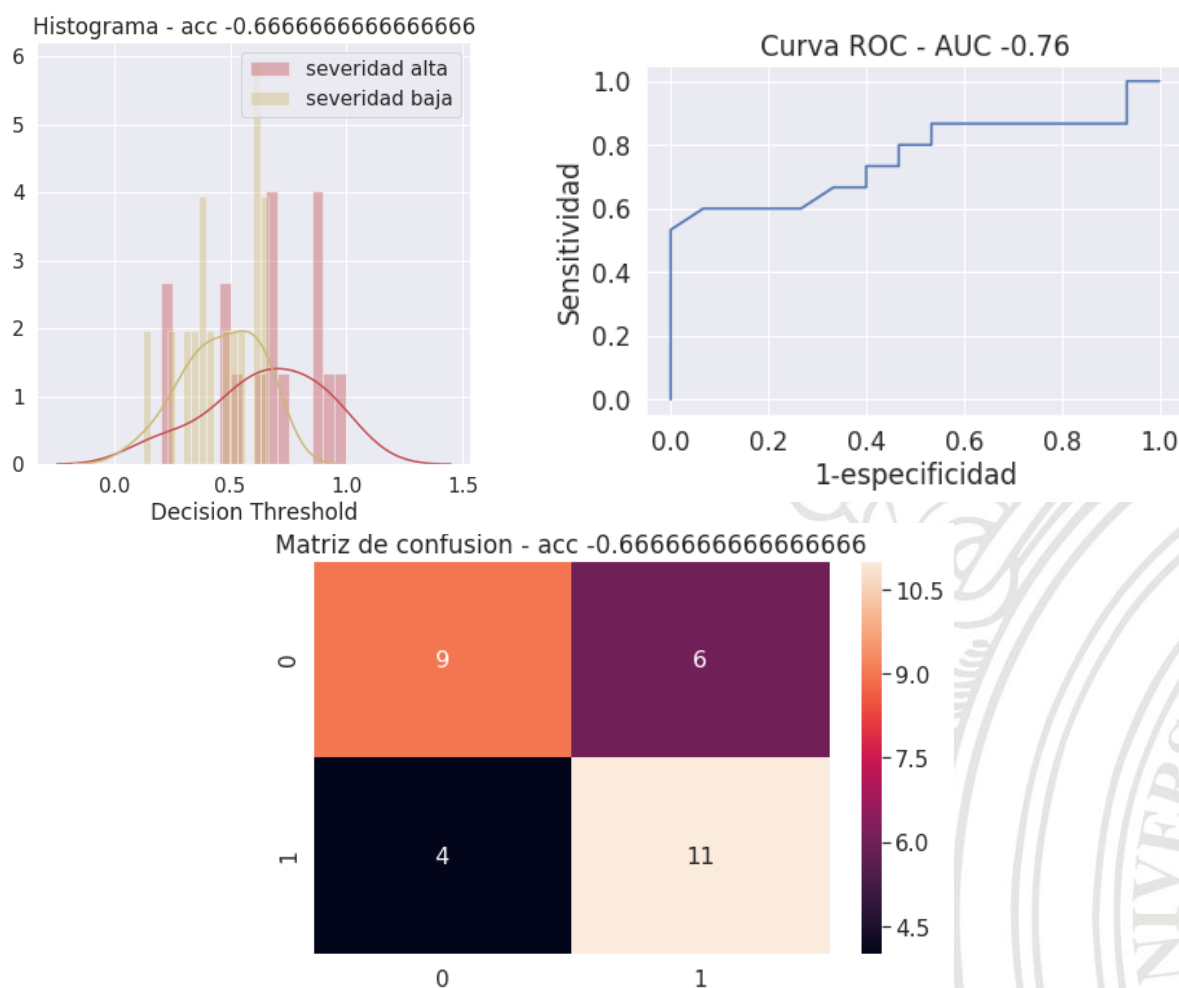


Figura 23. curva ROC y matriz de confusión de RF con las componentes principales

Se puede notar que con las componentes principales la curva ROC es similar pero más plana que la curva obtenida con 18 características, lo que indica que la clasificación tiene un rendimiento más bajo al aplicarle PCA a las características.

En la tabla 3 se muestra un resumen de los resultados obtenidos con estas técnicas

Técnica	Parámetros afinados	Tasa de aciertos	Especificidad	Sensitividad	AUC
SVM_Lineal	C=100	0.60	0.53	0.67	0.67
SVM_Lineal con PCA	C=1000	0.37	0.60	0.13	0.31
SVM_RBF	C=100, gamma=0.01	0.70	0.73	0.66	0.63

SVM_RBF con PCA	C=1000, gamma=0.01	0.40	0.53	0.26	0.33
RF	N Estimadores=16 Max profundidad=4 Min muestras =2	0.73	0.80	0.67	0.80
RF con PCA	N Estimadores=8 Max profundidad=4 Min muestras =2	0.66	0.6	0.73	0.76

Tabla 3. Resumen de pruebas y resultados para clasificación

De la tabla 2 se puede ver que los modelos entrenados con las componentes principales obtenidas tienen un desempeño mucho menor que aquellos entrenados con todas las características. Esto puede ser debido a que el algoritmo de PCA al momento de reducir las características les reduce la afectividad a los modelos, ya que se estarían entrenando con solo 2 características.

El mejor accuracy obtenido fue generado por la técnica RF, en general los resultados de esta técnica son bastante buenos, su histograma muestra una mejor separación entre las clases y la curva ROC es mas empinada que las otras técnicas.

Para clasificación el modelo de SVM entrenado con todas las características mostró un mejor desempeño general comparada a la SVM entrenada con las componentes principales. La SVM entrenada con kernel RBF tuvo un desempeño levemente mejor que en de la SVM entrenada con kernel lineal, esto es debido al uso de la Radio Basis Function (RBF), la cual introduce un nuevo parámetro que mapea las características a un nuevo espacio que ayuda a mejorar la clasificación y regresión.

4.3.2. SVR y RF para regresión

Luego de la afinación de parámetros y entrenamiento de la SVR, se usaron todos los datos para obtener el Error Medio Absoluto (MAE) y el coeficiente de correlación de rangos de spearman (ρ) del modelo usando toda la base de datos.

A continuación, se muestran los parámetros afinados y el gráfico de dispersión (Figura 24) de las etiquetas reales vs las etiquetas predichas.

Para kernel lineal, el parámetro C es 0.01 y épsilon es 1e-8. Se obtuvo un ρ de 0.14 y se obtuvo un MAE de 14.68.

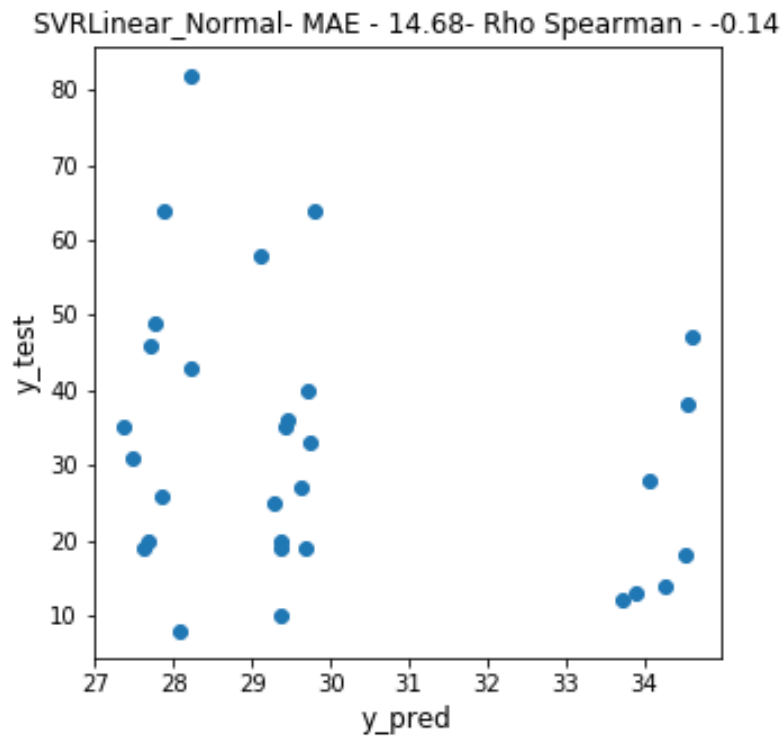


Figura 24. Gráfico de dispersión SVR lineal

Con las componentes principales, el parámetro C es 0.1 y épsilon es 0.1. Se obtuvo un ρ de 0.32 y se obtuvo un MAE de 14.95 (Figura 25).

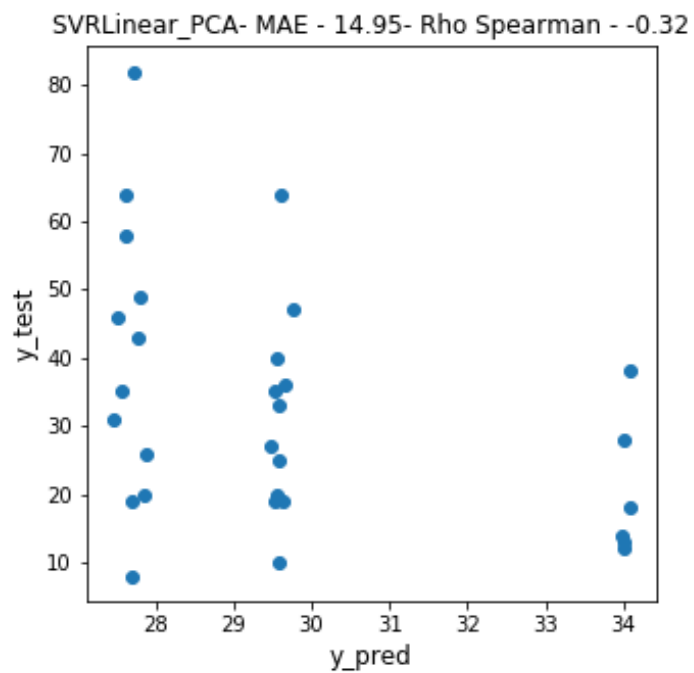


Figura 25. Gráfico de dispersión SVR lineal con las componentes principales

Se puede evidenciar que con el Kernel Lineal, la regresión por vectores de soporte genera una tendencia de predicción a 3 valores específicos, esto indica que el sistema está sub-entrenado y la regresión por esta técnica no es útil a las necesidades.

Para kernel RBF, el parámetro C es 100, gamma es 0.01 y épsilon es 1e-7. Se obtuvo un ρ de 0.23 y se obtuvo un MAE de 13.41 (Figura 26).

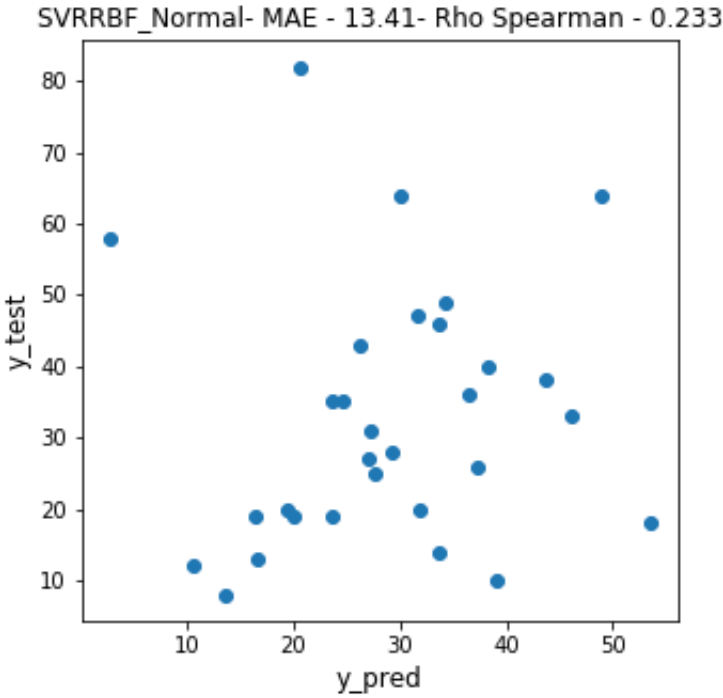


Figura 26. Gráfico de dispersión SVR RBF

Con las componentes principales, el parámetro C es 1000, gamma es 0.1 y épsilon es 0.1. Se obtuvo un ρ de 0.5 y se obtuvo un MAE de 11.53 (Figura 27).

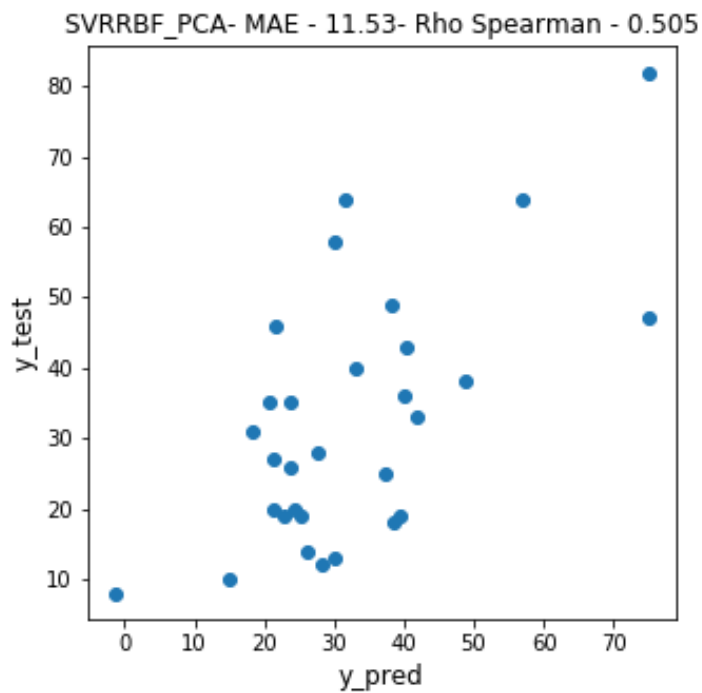


Figura 27. Gráfico de dispersión SVR RBF con las componentes principales

Para el modelo de SVR con kernel RBF se nota que el modelo entrenado con las componentes principales (Figura 27) tiene un mejor comportamiento que el modelo entrenado con todas las características (Figura 26), este último muestra un menor coeficiente de correlación de Spearman, un error medio absoluto más grande y mayor dispersión en la gráfica.

Los parámetros afinados para RF para regresión fueron, número de estimadores igual a 16, la máxima profundidad fue 4 y el número mínimo de muestras para división fue 2. Se obtuvo un ρ de 0.23 y se obtuvo un MAE de 12.98 (Figura 28).

RF_regressor_Normal- MAE - 10.98- Rho Spearman - 0.507

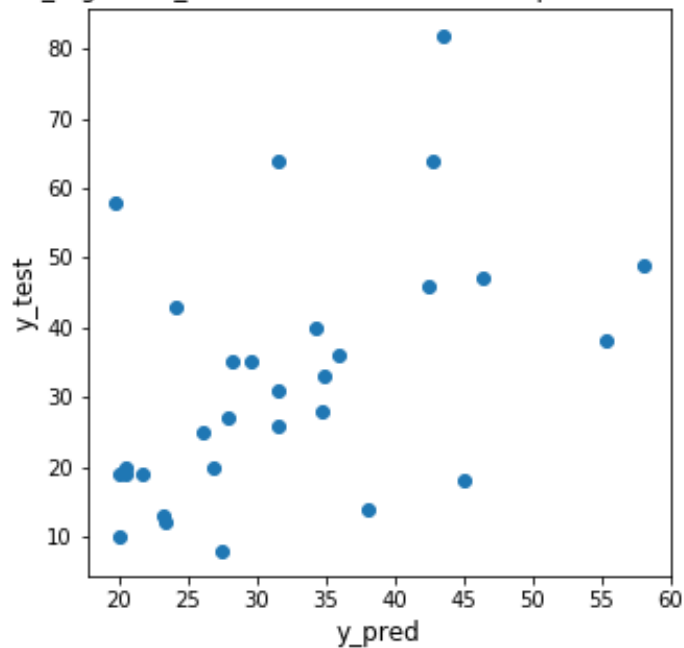


Figura 28. Gráfico de dispersión RF para regresión

Con las componentes principales el número de estimadores fue igual a 4, la máxima profundidad fue 16 y el número mínimo de muestras para división fue 8. Se obtuvo un ρ de 0.25 y se obtuvo un MAE de 14.99 (Figura 29).

RF_regressor_PCA- MAE - 14.99- Rho Spearman - -0.25

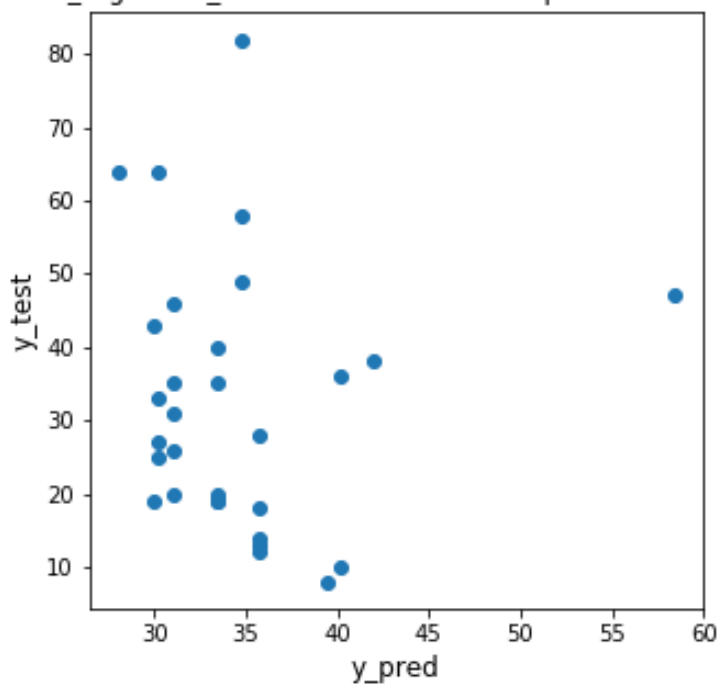


Figura 29. Gráfico de dispersión RF para regresión con las componentes principales

En los resultados del RF para regresión se puede ver en los gráficos (Figura 28 y 29) que hay gran dispersión en los datos, además de un muy bajo coeficiente de correlación de Spearman.

En la tabla 4 se muestra un resumen de los resultados obtenidos con estas técnicas

Técnica	Parámetros afinados	ρ	MAE
SVR_Lineal	C=0.01, epsilon= 1e-8	0.14	14.68
SVR_Lineal con PCA	C=0.1, epsilon= 0.1	0.32	14.95
SVR_RBF	C=100, gamma= 0.01, epsilon= 1e-7	0.32	13.41
SVR_RBF con PCA	C=1000, gamma= 0.1, epsilon= 0.1	0.50	11.53
RF para regresión	N Estimadores= 16 Max profundidad=4 Min muestras =2	0.51	10.98
RF para regresión con PCA	N Estimadores=4 Max profundidad=16 Min muestras =8	0.25	14.99

Tabla 4. Resumen de pruebas y resultados para regresión

En regresión los mejores modelos obtenidos fueron el de RF para regresión entrenado con todas las características, y la SVR con kernel RBF entrenada con las componentes principales. Sus coeficientes de correlación de Spearman fueron de 0.5. Aunque sus MAE son altos, tienen un valor aceptable teniendo en cuenta que se están tratando valores en el rango de 8-82.

En general se evidenció que con las características utilizadas la técnica de PCA para reducir dimensionalidad tiende a bajar sustancialmente el desempeño de los modelos de clasificación. En los modelos de regresión se presentó una mejoría cuando fueron entrenados con las componentes obtenidas. Ante una misma cantidad de información es mejor trabajar con más características en clasificación y menos característica en regresión.

Para regresión el método de SVR con kernel lineal tuvo un desempeño muy pobre, además de presentar inclinación a valores específicos, lo que afecta en gran medida el rendimiento del regresor, por lo que en este enfoque la técnica no resulta ser adecuada para la regresión.

Los resultados indican que las características utilizadas si aportan información sobre la severidad motora de los pacientes con la enfermedad de Parkinson según su grado de avance (índice UPDRS-Lower-Limbs). Pero la baja cantidad de las muestras de la base de datos genera variaciones en los resultados y hace que los modelos queden sub-entrenados, dando así un modelo con poca capacidad de generalización de los datos y variaciones en las diferentes pruebas de desempeño.

5. Conclusiones

La severidad motora de los pacientes con Parkinson puede ser evaluada por medios computacionales con un buen porcentaje de aciertos. Para esto debe realizarse un buen seguimiento del movimiento del paciente ya sea de forma automática o manual. Las características de Jitter, Shimmer y Energía del movimiento resultan ser útiles para la discriminación de los pacientes con la enfermedad de Parkinson por medio del índice UPDRS-Lower-Limbs. Ya que la información brindada por las el jitter, shimmer y energía del movimiento aporta información suficiente para el entrenamiento de modelos de aprendizaje de máquina que realicen la evaluación de la severidad motora de los pacientes con la enfermedad de parkinson.

Para el seguimiento automático del talón es necesario realizar una buena identificación del objeto de interés ya que el algoritmo de flujo óptico se encarga de realizar el seguimiento de una vecindad de pixeles previamente definidos, si estos pixeles no se detectan bien, el algoritmo no podrá realizar el seguimiento. Aunque se realice una buena detección del objeto no es garantía de un buen seguimiento ya que por cuestiones de la imagen es posible que se pierda el objeto en el proceso. Los resultados se pueden mejorar utilizando más información del sensor para generar un algoritmo de detección más robusto y mejorar las condiciones para el algoritmo de seguimiento.

En el presente trabajo con la información y técnicas utilizadas se concluye que para clasificación es bueno tener múltiples características ya que aumentan el hiper-espacio de las muestras, pero en regresión es mejor trabajar con la misma información en características reducidas, ya que simplifica el proceso. Esto indica que ante la misma cantidad de información es mejor trabajar con más características en clasificación y con menos características en regresión. Ya que en regresión se busca encontrar una función $f(x)$ que represente la distribución de datos, si esta función tiene múltiples variables de entrada, se genera un mayor riesgo de error.

La técnica PCA, aunque se fundamenta en aumentar la varianza del espacio de características generando componentes que contengan la información

representada de forma diferente, es posible que pueda ser contraproducente ya que la reducción de características puede mejorar el costo computacional, pero reducir el desempeño del sistema. Se hace necesario comparar la reducción del costo computacional con la reducción del desempeño y evaluar si es conveniente aplicarlo.

La base de datos a utilizar en técnicas de aprendizaje de máquina como las utilizadas en el presente trabajo debe ser de un tamaño conveniente dependiendo del problema, ya que, si el tamaño de la base de datos es bajo, el modelo puede quedar sub-entrenado y no alcanzará buena generalización del entrenamiento.

Tanto para clasificación como para regresión los mejores resultados se obtienen usando un algoritmo basado en bosques aleatorios. Este resultado indica que para el presente trabajo el uso del promedio en caso de regresión o voto por mayoría en caso de clasificación, de los diferentes resultados de los árboles de decisión genera un resultado más robusto y acertado que el de una función obtenida por los vectores de soporte.

6. Líneas futuras

Realizar una buena afinación del algoritmo de detección y seguimiento del talón con el fin de poder obtener una señal automática y evitar el error humano que se pueda presentar en el seguimiento manual del objeto, con el fin de automatizar el proceso y poder generar una herramienta que sirva de apoyo en la evaluación de la severidad motora de un paciente sin necesidad de intervención humana.

Ampliar la base de datos de trabajo tomando más muestras de pacientes diagnosticados con la enfermedad de Parkinson con el fin de generar modelos que no tengan dependencia al conjunto de entrenamiento y no queden sub-entrenados ni sobre entrenados, es decir, que tengan buena capacidad de generalización, esto con el fin de obtener un sistema con un mejor desempeño, una desviación estándar baja y mayor confiabilidad en los resultados.

Continuar la línea de investigación utilizando otras características como la velocidad promedio del movimiento, o características obtenidas por técnicas de aprendizaje profundo como redes neuronales convolucionales y recurrentes. También se puede utilizar otras tareas de análisis que involucran otros planos de movimiento, las cuales se encuentran en la base de datos utilizada en el presente trabajo.

Referencias Bibliográficas

- [1] LÓPEZ-PABÓN, Felipe O.; ARIAS-VERGARA, Tomas; OROZCO-ARROYAVE, Juan R. Cepstral analysis and Hilbert-Huang transform for automatic detection of Parkinson's disease. *Tecnológicas*, 2020, vol. 23, no 47, p. 91-106.
- [2] RIOS-URREGO, Cristian D., et al. Analysis and evaluation of handwriting in patients with Parkinson's disease using kinematic, geometrical, and non-linear features. *Computer methods and programs in biomedicine*, 2019, vol. 173, p. 43-52.
- [3] FARRÚS, Mireia; HERNANDO, Javier; EJARQUE, Pascual. Jitter and shimmer measurements for speaker recognition. En *Eighth annual conference of the international speech communication association*. 2007.
- [4] JAKKULA, Vikramaditya. Tutorial on support vector machine (svm). *School of EECS, Washington State University*, 2006, vol. 37.
- [5] AWAD, Mariette. KHANNA, Rahul. *Efficient Learning Machines. Support Vector Regression*. New York: Apress, 2015. 67p.
- [6] MARTÍNEZ ORTEGA, Rosa María, et al. El coeficiente de correlación de los rangos de Spearman caracterización. *Revista Habanera de Ciencias Médicas*, 2009, vol. 8, no 2, p. 0-0.
- [7] GOETZ, Christopher G., et al. Movement Disorder Society-sponsored revision of the Unified Parkinson's Disease Rating Scale (MDS-UPDRS): scale presentation and clinimetric testing results. *Movement disorders: official journal of the Movement Disorder Society*, 2008, vol. 23, no 15, p. 2129-2170.
- [8] VALLE BENAVIDES, Ana Rocío del. *Curvas ROC (Receiver-Operating-Characteristic) y sus aplicaciones*. 2017.
- [9] WOLD, Svante; ESBENSEN, Kim; GELADI, Paul. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 1987, vol. 2, no 1-3, p. 37-52.
- [10] BREIMAN, Leo. Random forests. *Machine learning*, 2001, vol. 45, no 1, p. 5-32.
- [11] BROWNE, Michael W. Cross-validation methods. *Journal of mathematical psychology*, 2000, vol. 44, no 1, p. 108-132.

- [12] PALMA ROLDÁN, Carlos Fernando. Human movements evaluation using depth sensors. 2017.
- [13] OCHOA, C. Osorio, et al. Análisis de marcha mediante plataforma Kinect. *La Investigación y las Aplicaciones en Ciencias de la Computación. FCC-BUAP. Puebla, México, 2011*, p. 17-21.
- [14] MATHE, Ladislao; SAMBAN, Darío; GÓMEZ, Gabriel. Estudio del funcionamiento del sensor Kinect y aplicaciones para bioingeniería. *Proyecto final de carrera de Ingeniería en Computación, Universidad Nacional de Córdoba, Córdoba, Veracruz, 2012*.
- [15] KHOSHELHAM, Kouros. Accuracy analysis of kinect depth data. En *ISPRS workshop laser scanning*. 2011.
- [16] PREIS, Johannes, et al. Gait recognition with kinect. En *1st international workshop on kinect in pervasive computing*. New Castle, UK, 2012. p. 1-4.
- [17] DROR, Ben, et al. Automatic assessment of Parkinson's Disease from natural hands movements using 3D depth sensor. En *2014 IEEE 28th Convention of Electrical & Electronics Engineers in Israel (IEEEI)*. IEEE, 2014. p. 1-5.
- [18] VINOKUROV, Nomi, et al. Quantifying hypomimia in parkinson patients using a depth camera. En *International Symposium on Pervasive Computing Paradigms for Mental Health*. Springer, Cham, 2015. p. 63-71.
- [19] GALNA, Brook, et al. Accuracy of the Microsoft Kinect sensor for measuring movement in people with Parkinson's disease. *Gait & posture*, 2014, vol. 39, no 4, p. 1062-1068.
- [20] BIGY, A. Amini Maghsoud, et al. Recognition of postures and Freezing of Gait in Parkinson's disease patients using Microsoft Kinect sensor. En *2015 7th International IEEE/EMBS Conference on Neural Engineering (NER)*. IEEE, 2015. p. 731-734.
- [21] MARTÍNEZ-FERNÁNDEZ, Raúl, et al. Actualización en la enfermedad de Parkinson. *Revista Médica Clínica Las Condes*, 2016, vol. 27, no 3, p. 363-379.
- [22] VÁSQUEZ-CORREA, Juan Camilo, et al. Multimodal assessment of Parkinson's disease: a deep learning approach. *IEEE journal of biomedical and health informatics*, 2018, vol. 23, no 4, p. 1618-1630.

[23] ZENG, Wei, et al. Parkinson's disease classification using gait analysis via deterministic learning. *Neuroscience letters*, 2016, vol. 633, p. 268-278.

[24] OH, Jeonghoon, et al. Comparison of predicted kinetic variables between Parkinson's disease patients and healthy age-matched control using a depth sensor-driven full-body musculoskeletal model. *Gait & posture*, 2020, vol. 76, p. 151-156.

[25] PROCHÁZKA, Aleš, et al. Bayesian classification and analysis of gait disorders using image and depth sensors of Microsoft Kinect. *Digital Signal Processing*, 2015, vol. 47, p. 169-177.

[26] ŤUPA, Ondřej, et al. Motion tracking and gait feature estimation for recognising Parkinson's disease using MS Kinect. *Biomedical engineering online*, 2015, vol. 14, no 1, p. 1-20.

