Trabajo Final de TMD

Reconocimiento de actividad con smartphones

Alumno: Alonso Pablo

1. Introducción

Este dataset fue construido a partir de observar las actividades de 30 individuos con edad entre 19 y 48 años realizando actividades mientras llevaban smartphones (Samsung Galaxy S II) con sensores inerciales integrados (acelerómetro y giroscopio). Se capturaron la acelaración lineal y la velocidad angular en un radio constante de 50Hz.

Las señales de los sensores (acelerómetro y giroscopio) fueron pre-procesados aplicando filtros de ruido y sampleadas en ventanas de 2.56 segundos y una superposición del 50% (128 lecturas por ventana). Por cada ventana, se obtiene un vector de datos aplicando operaciones como media,desviación estándar,máximo y mínimo, etc.

Las activades fueron: caminar, subir escaleras, bajar escaleras,sentarse, pararse,acostarse. El problema es poder indentificar que actividad estaba realizando un individuo al momento de hacer la medición con los artefactos previamente descriptos.

Cada observación del dataset contiene:

- Aceleración triaxial (medida desde el acelerómetro) y la aceleración estimada del cuerpo.
- Velocidad angular triaxial (medida desde el giroscopio).
- En total son 561 features con variables de dominio de tiempo y frecuencia, más la etiqueta de la actividad y el ID del usuario.

2. Pre-análisis

En total el dataset tiene 563 features . No contiene campos con valores nulos o desconocidos y los features están normalizados en el rango [-1,1] . Para hacer el análisis se descarta el feature id de usuario porque se considera que no aporta ninguna información relevante al problema.

Balance de las clases:

LAYING	SITTING	STANDING
18.9	17.3	18.5
WALKING	WALKING DOWNSTAIRS	WALKING UPSTAIRS
16.7	13.7	15

No se observa un desbalance significativo entre las clases por lo que no resulta necesario aplicar técnicas de sampleo.

2. Visualización de variables

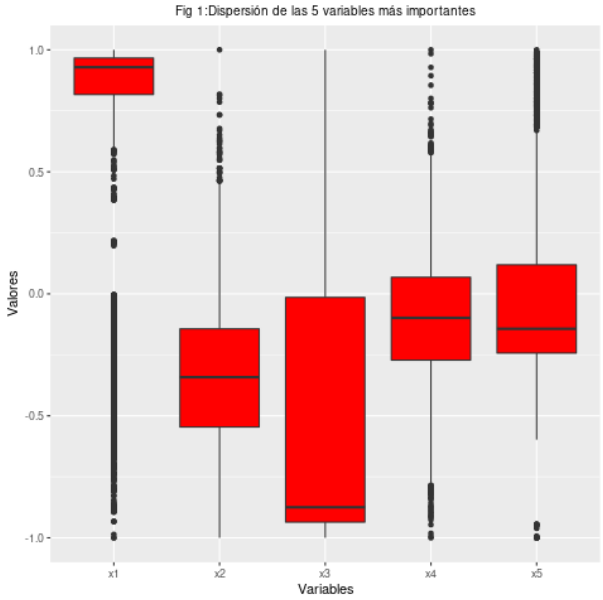
Dado el número de features resulta imprescindible usar algún método que permita reducir la dimensionalidad de los datos y analizar solo aquellas variables que sean importantes. Se utiliza Recursive Feature Elimination junto con Random Forest para determinar la importacia las variables. Se usó este algoritmo porque considera la relevancia de las variables cuando trabajan en conjunto. Se listan las 5 variables más relevantes determinadas por el algoritmo:

Ranking de variables
x1:=tGravityAcc-min-X
x2:=tGravityAcc-arCoeff-Y
x3:=tBodyAcc-max-X
x4:=fBodyGyro-meanFreq-X
x5:=tGravityAcc-mean-Y

El algoritmo dice que los features más influyentes para determinar la actividad de los individuos fueron:

- Aceleración de la gravedad (mínimo) en el eje x.
- Aceleración de la gravedad (mínimo) en el eje y.
- Aceleración del cuerpo (máximo) en el eje x.
- Giro del cuerpo (frecuencia media) en el eje x.
- Aceleración de la gravedad (coeficiente de autoregresión) en el eje y.

2.1 Análisis univariado



En **Fig 1** se observa la dispersión de las 5 variables más relevantes El boxplot permite ver dónde se encuentran el rango intercuartílico y los outliers.

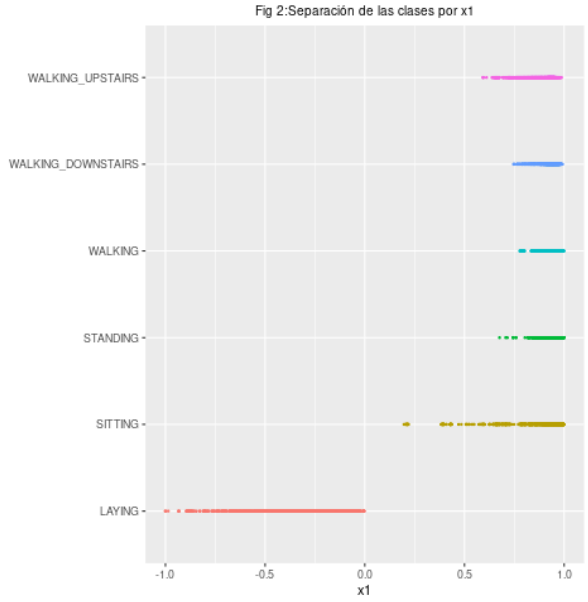


Fig 2: Cuando el sujeto está acostado no hay movimiento y por lo tanto no hay aceleración de un movimiento hacia abajo, de ahí que los valores en ese caso sean nulos o menores que cuando se está realizando otra actividad.

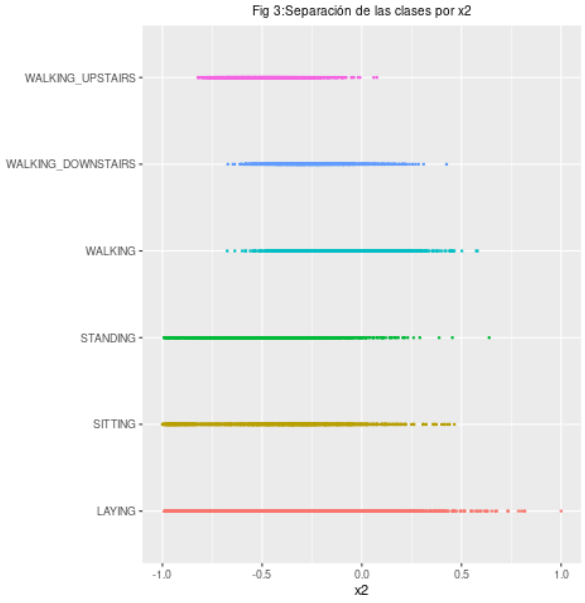


Fig 3: No hay una clara separación entre clases por parte de x2, aunque hay una clara diferencia en como se concentran los valores si el sujeto se está desplazando.

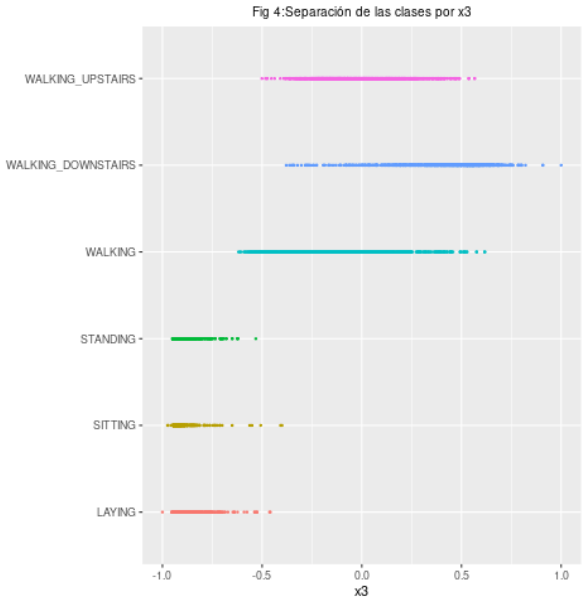


Fig 4: Claramente la aceleración del cuerpo es mayor cuando el sujeto se está desplazando.

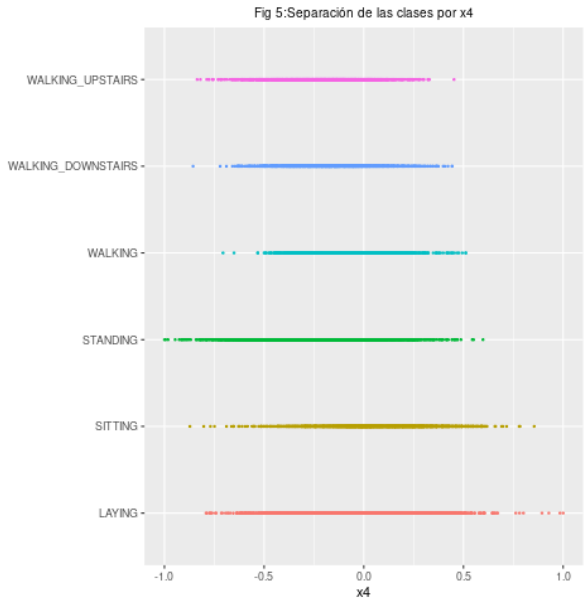


Fig 5: No parece aportar demasiada información a la actividad del sujeto

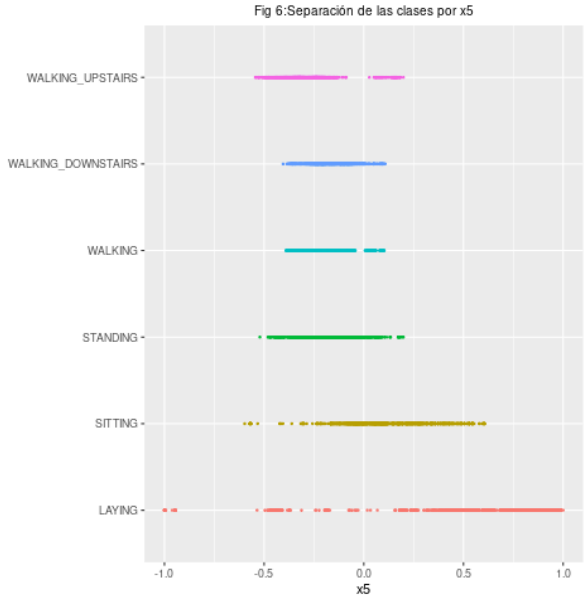


Fig 6: x5 parece tener una fuerte relacion con el hecho de si el individuo está acostado. En este caso, la variable toma valores mayores comparando con las otras actividades.

2.2 Análisis bivariado

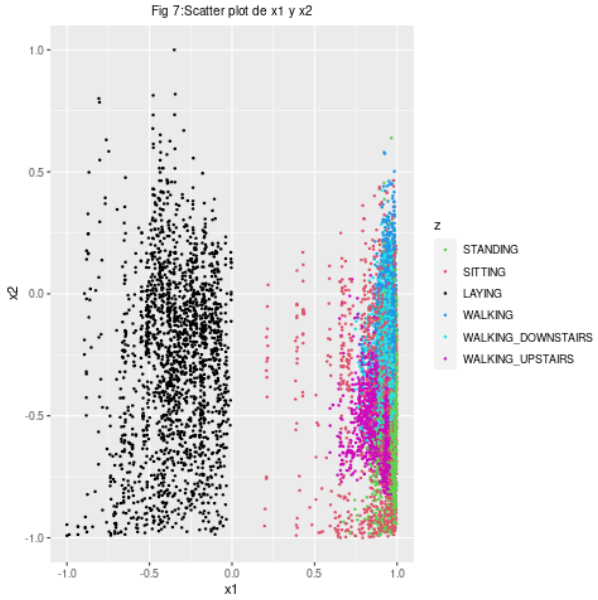


Fig 7:Como ya se vio,x1 determina si el individuo está acostado, y en algunos casos si está sentado. Para el resto de las clases, estas 2 variables no aportan mucha información.

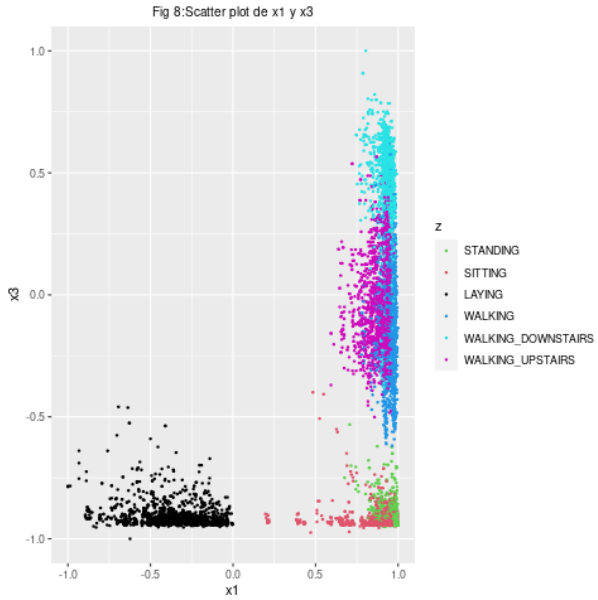


Fig 8:Nuevamente,x1 determina si el individuo está acostado. Hay una clara separación entre las clases que indican que el sujeto se está desplazando y las

que no. En ciertas regiones hay clases predominantes sobre otras, por lo que se concluye que estas 2 variables trabajan bien en conjunto y aportan mucha información al problema.

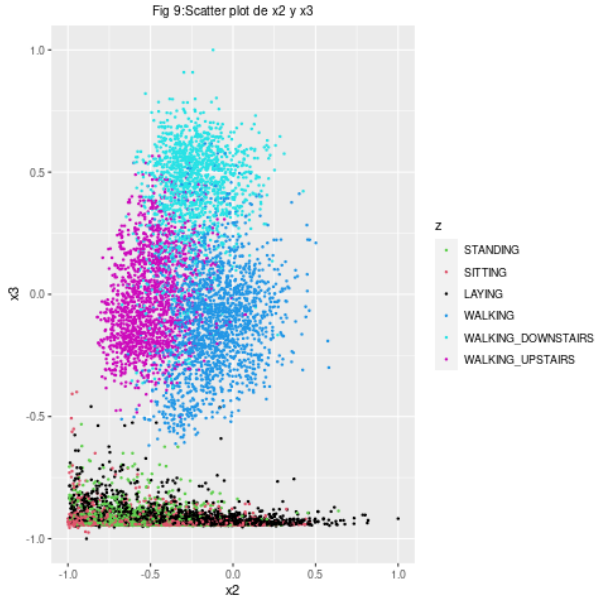


Fig 9:Hay una clara separación sobre si el sujeto se está desplazando o no. Además definen de manera bastante precisa si el individuo camina o sube o baja escaleras. Para el resto de las clases se observa solapamiento.

2.3 Análisis trivariado

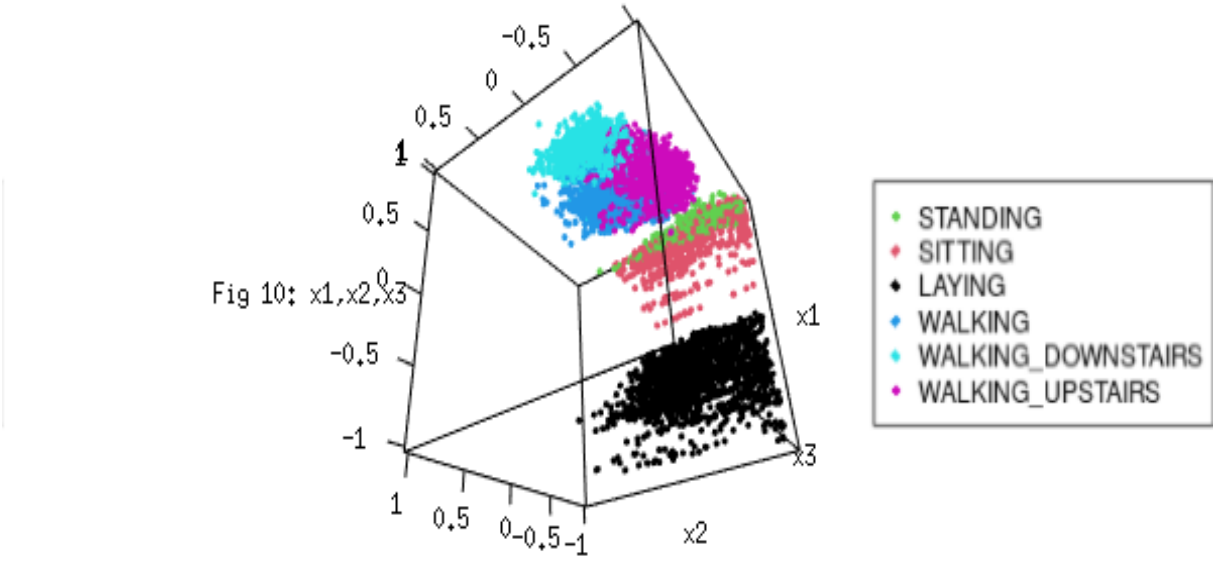


Fig 10: x1,x2,x3

Fig 10: Hay una separación muy evidente entre LAYING y el resto de las clases, como ya se había visto en figuras anteriores debido a x1. Ahora también hay una buena separación entre SITTING y STANDING con respecto a las clases en las que hay desplazamiento gracias a la variable x3 (aceleración del cuerpo). Hay un poco de solapamiento entre SITTING y STANDING si x3 es cercano a -1 (menor aceleración), pero el solapamiento disminuye a medida que x3 es mayor a -1 (mayor aceleración).

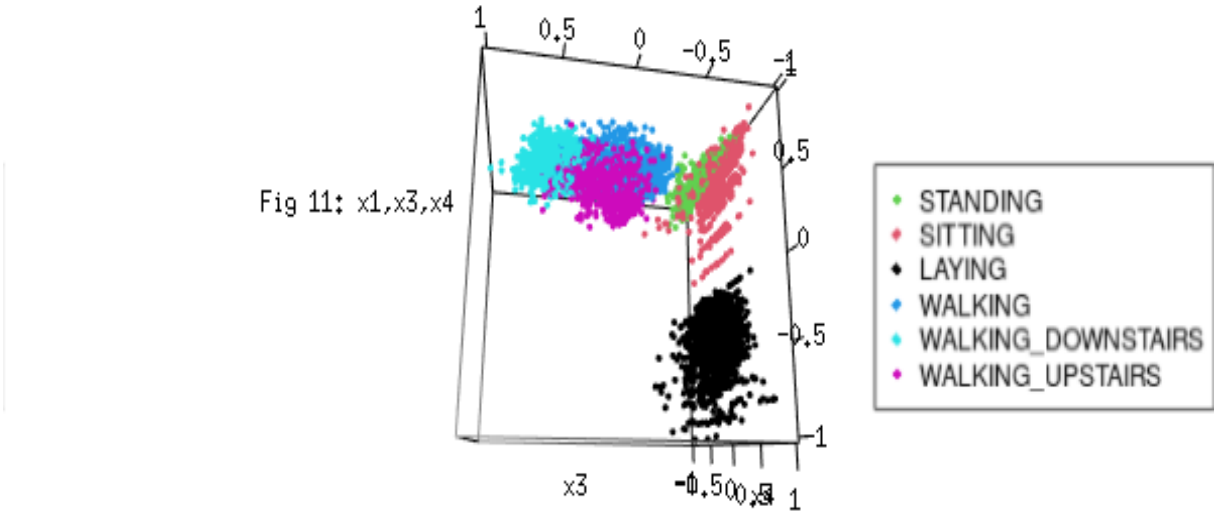


Fig 11: En este caso la variable x4 (giro del cuerpo) parece determinar mejor que en el caso anterior si el sujeto se está sentando o se está parando. Comparado con **Fig 10**, hay más solapamiento entre las clases que indican desplazamiento del sujeto.

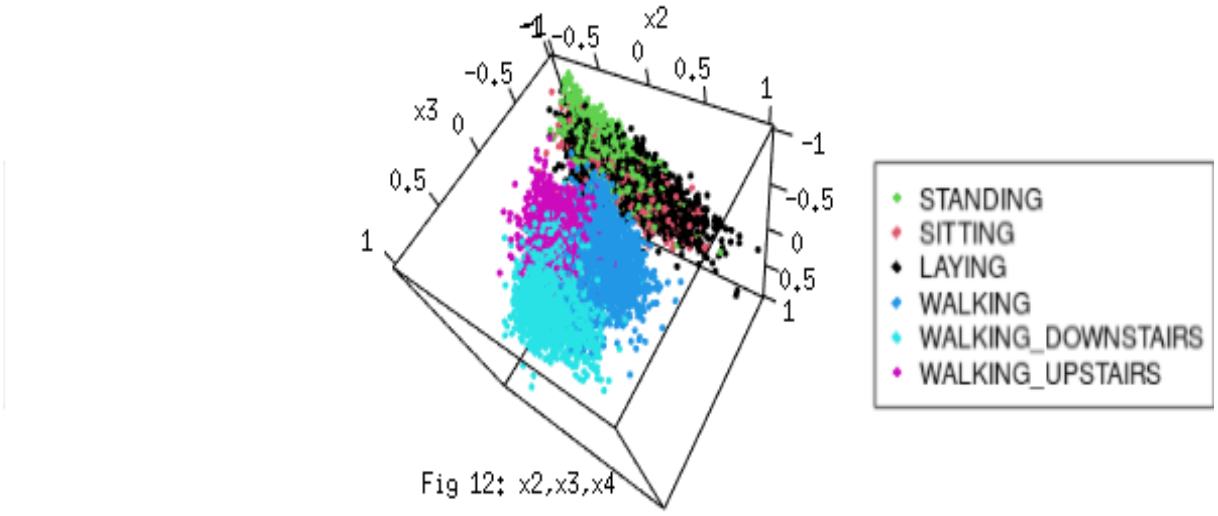


Fig 12: Nuevamente hay una clara separación entre las clases que indican desplazamiento y las que no. Ahora hay mucho solapamiento entre las clases STANDING,SITTING y LAYING. Estas 3 variables distinguen muy bien que

tipo de desplazamiento se está produciendo.

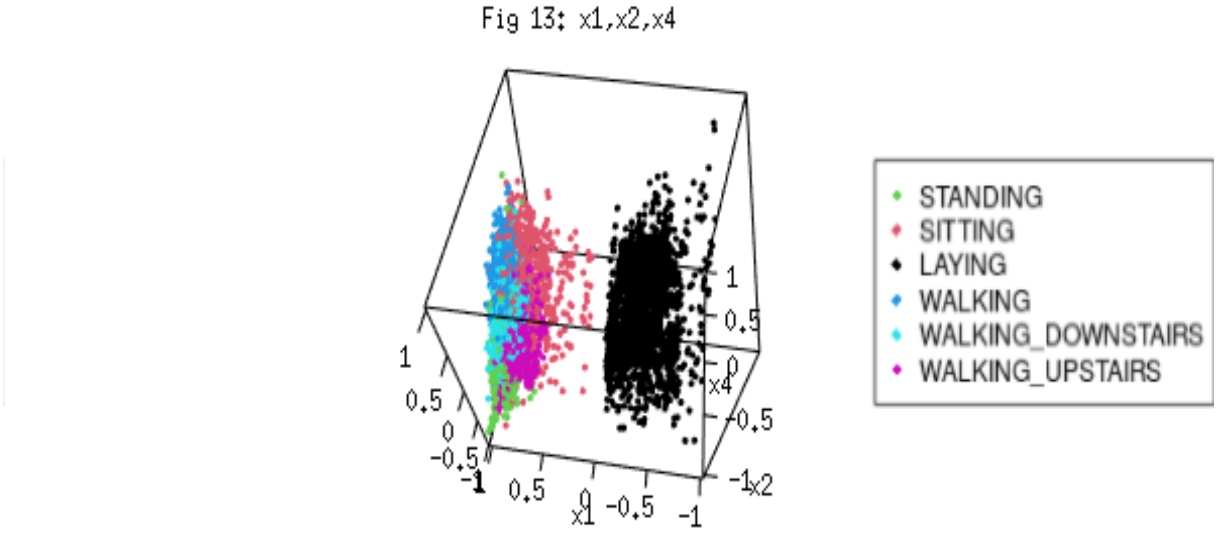


Fig 13: En este caso la única separación clara es entre LAYING y el resto de las clases, sin embargo para el resto de las clases hay mucho solapamiento.

3. Clustering

3.1 Agrupamiento con k-means

Table 1: Kmeans con k = 6 y 280 dimensiones

	3	6	5	4	1	2
LAYING	1917	0	4	0	0	23
SITTING	50	1139	579	0	0	9
STANDING	0	946	958	0	0	2
WALKING	0	0	0	1390	239	93
WALKING_DOWNSTAIRS	0	0	0	250	1009	147
WALKING_UPSTAIRS	0	0	0	41	123	1380

Una solución de haber ejecutado k-means con k = 6 se muestra en Tabla 1, se observa que el algoritmo pone un centro en la clase LAYING, 2 centros compartidos por las clases SITTING y STANDING y el resto de los centros uno en el centro de cada clase restante.

Table 2: Kmeans con k = 6 y 50 dimensiones

	6	1	4	2	3	5
LAYING	1940	0	0	3	0	1
SITTING	17	570	626	564	0	0
STANDING	0	4	1105	789	3	5
WALKING	0	0	65	14	869	774
WALKING_DOWNSTAIRS	0	0	0	0	880	526
WALKING_UPSTAIRS	0	0	0	56	607	881

Tabla 2 muestra una solución de 6 means luego de haber bajado la dimensionalidad, se observa más solapamiento entre las clases al haber reducino el número de dimensiones.

Table 3: HClust con 280 dimensiones

	3	5	1	2	4	6
LAYING	50	0	1893	0	0	1
SITTING	0	0	1777	0	0	0
STANDING	0	0	1906	0	0	0
WALKING	0	0	0	1721	1	0
WALKING_DOWNSTAIRS	0	2	0	1381	23	0
WALKING_UPSTAIRS	0	0	0	1544	0	0

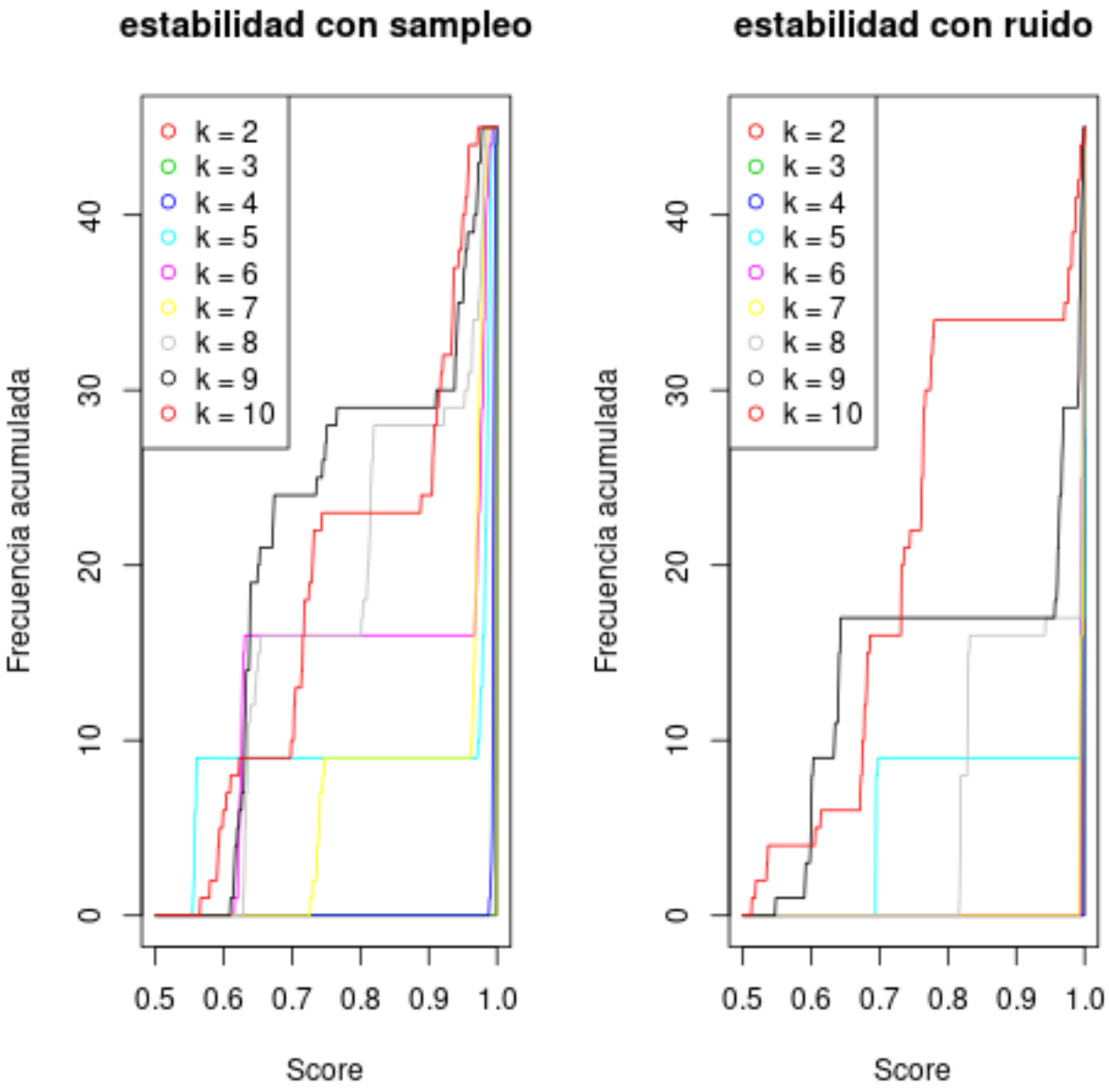
Table 4: HClust con 50 dimensiones

	2	3	1	4	5	6
LAYING	1808	85	0	51	0	0
SITTING	58	4	1711	0	1	3
STANDING	0	0	1906	0	0	0
WALKING	0	0	1722	0	0	0
WALKING_DOWNSTAIRS	0	0	1406	0	0	0
WALKING_UPSTAIRS	0	0	1544	0	0	0

Table 5: Hclust con 280 dimensiones

	3	6	1	2	4	5
LAYING	39	0	1905	0	0	0
SITTING	2	0	1774	1	0	0
STANDING	0	0	1906	0	0	0
WALKING	0	0	0	1706	1	15
WALKING_DOWNSTAIRS	0	2	0	1333	33	38
WALKING_UPSTAIRS	0	0	0	1490	0	54

3.2 Estabilidad de los clusters



Usando sampleo y ruido para modificar el dataset original, con el algoritmo de estabilidad se determina que el número óptimo de clusters es 5.

4. Clasificación

Se usarán 2 métodos distintos : random forest y SVM con kernel polinómico. Para entrenar los modelos se usan un subconjunto de n variables tomadas del ranking obtenido de la sección 2. Luego se podrá comparar la cantidad de variables que realmente aportan información para la clasificación. Las medidas de error se obtienen promediando los resultados obtenidos con 5-folds.

Resultados obtenidos con random forest:

Número de variables empleado	Error
5	0.06
50	0.02
100	0.02
300	0.02
400	0.02
561	0.02

No se obtienen mejoras considerables al entrenar random forest con más de 50 variables. Posiblemente el algoritmo esté limitando la cantidad de variables o podando los árboles del ensamble para no sobreajustar los resultados.

Resultados obtenidos con SVM:

Número de variables empleado	Error
5	0.1
50	0.03
100	0.03
300	0.02
400	0.014
561	0.014

Se observan mejoras significativas en los resultados a medida que se incrementa el número de variables tomadas. Dado que los resultados no mejoran a partir de haber tomado 400 variables, se concluye que al menos las últimas 161 variables no aportan información nueva al algoritmo que le permita mejorar su decisión.