Universidad de Granada

Proyecto final:

Aprendizaje Automático

Parkinson Telemonitoring

Autor:
Luis Suárez Lloréns

28 de junio de 2016

Índice

1. Motivación **2**

2. Información de los datos **3**

3. Selección de modelos **4**

4. Manipulación de los datos **6**
4.1. Codificación de los datos 6
4.2. Normalización 6
4.3. Selección de características 7

5. Desarrollo **8**
5.1. Modelos paramétricos 8
5.2. Modelo no paramétrico 8

6. Resultados **10**
6.1. Resultados anteriores 10
6.2. Resultados de modelos paramétricos 10
6.3. Resultados de modelo no paramétrico 11

7. Bibliografía **12**

1. Motivación

La medicina se ha visto impulsada en los últimos años por los avances técnicos. En concreto, se han utilizado técnicas de inteligencia artificial para la ayuda en el diagnóstico y seguimiento de enfermedades.

El objetivo no es sustituir la función de los profesionales del campo de la medicina, si no crear herramientas para mejorar el tratamiento a los pacientes.

Este trabajo se centra en el seguimiento de la enfermedad de Parkinson, y en obtener una estimación del grado en la que esta afecta al paciente de manera automática, mediante la medición de la voz del mismo. Además, estas mediciones se pueden realizar de forma automática en la casa del paciente.

Todo esto nos permitiría realizar un mejor seguimiento del paciente, y reduciendo posibles inconvenientes, como los desplazamientos por parte del paciente a su centro médico y la realización de pruebas médicas más costosas, tanto económicamente como en tiempo de personal.

2. Información de los datos

La base de datos, creada por los investigadores Athanasios Tsanas y Max Little de la universidad de Oxford en colaboración con varios centros médicos de los Estados Unidos, contiene datos relativos al habla de varios enfermos de Parkinson.

En la creación de la base de datos, 42 pacientes fueron monitorizados en un periodo de 6 meses. En total, se realizaron 5,875 mediciones a los pacientes.

En la investigación se trata de conocer el grado de Parkinson del paciente, utilizando para ello la escala unificada para la evaluación de la enfermedad de Parkinson (UPDRS). Por tanto, el objetivo es utilizar los 16 valores obtenidos de la medición del habla para estimar los valores de las variables "motor UPDRSτ "total UPDRS".

En la tabla de los datos también se puede puede consultar el identificador del paciente al que se realizó la medición, su edad, género y el momento exacto de su realización, pero el estudio original ignora estos factores.

Se puede consultar más información en el artículo original [1].

3. Selección de modelos

Ya hemos visto que el objetivo de este estudio es estimar dos valores reales. Nos centraremos para la selección de modelos en los métodos de regresión que hemos estudiado durante la asignatura.

Utilizaremos como una primera aproximación la regresión logística. Esta técnica, básica en el desarrollo del aprendizaje automático, sigue siendo de validez y puede obtener buenos resultados. Además, dada la simplicidad de la técnica, podemos ajustar varios modelos —lineales, cuadráticos, ...— y utilizar estos resultados como medidas para comparar los resultados obtenidos con técnicas más complejas.

Utilizaremos 4 modelos lineales para comparar resultados. El primero, considerará todas las variables. Luego utilizaremos dos modelos con menos variables, utilizando nuestra selección de características. Por último, utilizaremos la mejor selección de características del artículo de referencia de nuestra base de datos [1].

No consideramos el problema de la selección del mejor de los cuatro modelos, pues queremos realizar una comparación con los resultados obtenidos en el artículo de A. Tsanas [2]. De querer seleccionar de una manera apropiada el mejor modelo, no podríamos usar el mejor resultado obtenido en la sección de resultados, pues estaríamos tomando el modelo que mejor se adapta a esos datos de test concretos. Sería más apropiado realizar una validación cruzada.

Otra aproximación posible es usar un modelo de función de base radial. Parece lógico pensar que dos mediciones con valores parecidos nos darán valores de UPDRS parecidos. Para poder usar esta técnica, además de la selección del núcleo, tenemos que preocuparnos de manipular los datos, normalizándolos, pues los resultados de las mediciones no se encuentran todos en la misma escala.

Utilizaremos, a modo de comparación, dos modelos distintos. El primero, utilizará todas las variables de la base de datos, y el segundo una selección de los mismos.

Por último, utilizaremos una modelo de red neuronal multicapa. La po-

tencia y principal bondad de este tipo de modelos es su capacidad de aprender automáticamente posibles relaciones ocultas de gran complejidad de oculta detrás de los datos. Podríamos entender esto como una especie de autoselección de características. Esto es importante en el ámbito de la medicina, donde los parámetros están interconectados, posiblemente mediante procesos muy complejos de muy difícil modelización en ecuaciones. El problema de usar este tipo de técnicas es la selección de la arquitectura —número de neuronas, distribución de las neuronas en capas, tipo de funciones no lineales, ...—.

4. Manipulación de los datos

4.1. Codificación de los datos

Los datos obtenidos directamente se encuentran en una codificación perfecta para empezar a trabajar. Se tratan de 18 columnas —una vez eliminadas las columnas de identificador, edad,...— de valores reales, dos de ellas son los valores a estimar, y el resto las mediciones de la voz de los pacientes.

Reservaremos un 20 % aleatorio de la muestra para test, siendo el resto de la base de datos usada para el entrenamiento y la selección de modelos y parámetros.

4.2. Normalización

Vamos a necesitar normalizar para aplicar algunos modelos —más información en la sección de desarrollo—. Es un proceso sencillo, pero que hay que realizar con cuidado para no contaminar el conjunto de test y el experimento en general.

Destacar que no vamos a normalizar las variables de salida. Esto hará que sean directamente comparables los resultados de los métodos que usen la base de datos normalizada con los que la usen sin normalizar.

El proceso utilizado es el siguiente:

- Normalizar los datos de entrenamiento.
- Guardar los parámetros de la normalización de entrenamiento —media y escala—.
- Normalizar los datos de test con los parámetros de normalización de entrenamiento.

Utilizando estos pasos, no utilizamos información del test para realizar la normalización, pero ambos conjuntos quedan perfectamente normalizados para su uso posterior.

4.3. Selección de características

Para la selección de características, utilizaremos una regresión LASSO, como vimos en prácticas. Este modelo de regresión, aplica una regularización durante el cálculo de los parámetros. Tiene la propiedad de llevar a 0 los valores menos significativos, produciendo una buena selección de variables.

Tras normalizar los datos, aplicamos esta regresión y obtenemos los coeficientes. A mayor valor tenga el coeficiente, de mayor importancia será para realizar la regresión. Si tomamos un umbral, y seleccionamos los coeficientes en valor absoluto mayores que dicho umbral, obtendremos los coeficientes más importantes.

Usando dicha técnica, consideramos dos conjuntos de características:

- C_1: Jitter... + Jitter.Abs. + Shimmer.APQ5 + Shimmer.APQ11 + HNR + DFA + PPE
- C_2: Shimmer.APQ5 + Shimmer.APQ11 + HNR + DFA + PPE

Definimos, además, C_t como el conjunto de todas las características y C_a como el mejor conjunto de características del artículo original —HNR + RPDE + DFA + PPE—.

5. Desarrollo

5.1. Modelos paramétricos

No queda mucho que comentar en esta sección. Simplemente, entrenamos los 4 modelos con los datos de entrenamiento, con la función de R *glm*.

Tras esto, el modelo ya está preparado para ser utilizado para predecir valores. Se usarán en la sección de resultados para predecir los valores de test y obtener el error de cada uno.

5.2. Modelo no paramétrico

Aquí sí encontramos grandes complicaciones, desde el punto de vista computacional.

Los dos modelos que se pueden elegir, K-NN o función de base radial, no aprenden de ninguna forma. Simplemente, almacenan los datos. Tras esto, para cada dato que queremos predecir, se tienen que calcular las distancias a todos los puntos, y en función de las mismas, actuar. En el caso del K-NN tomando las K más cercanas y en el caso de la función de base radial, sirviendo como peso para una media ponderada.

Esto hace que el proceso de cálculo de una predicción sea un proceso muy lento.

En cuanto al código de predicción de la función de base radial, no hemos encontrado módulos que realicen exactamente esto. Hemos encontrado un código que realiza redes de funciones de base radial en el paquete de R *RSNNS*, pero no es exactamente lo que se busca.

Por tanto, hemos realizado el código a mano. El algoritmo es el siguiente:

- Para cada dato a predecir Y:
    - Calcular el peso de cada dato de entrenamiento X, usando $\varphi(\text{dist}(X, Y) * \alpha)$, siendo $\varphi(x)$ la función gaussiana.
    - Realizamos la media ponderada como $\frac{\sum_i \text{peso}_i \times \text{valor}_i}{\sum_i \text{dist}_i}$.

El proceso de selección de α y del modelo que consideraremos en los resultados, debería realizarse mediante validación cruzada. Sin embargo, como ya hemos comentado, este proceso sería muy costoso en tiempo de computo de hacer correctamente, pudiendo tardar varios días en obtener una respuesta.

Por tanto, como simplificación para intentar sortear este problema, tomaremos una muestra del conjunto de entrenamiento y realizaremos el entrenamiento con el resto de valores. De alguna manera, sería solo realizar uno de los pasos de una validación cruzada.

No es el mecanismo oportuno como ya hemos comentado, pero lo usaremos para sortear las limitaciones técnicas que hemos encontrado.

El mejor resultado lo encontramos finalmente con el conjunto completo de variables y un α de 5.

	Motor UPDRS	Total UPDRS
C_t	6.547839	8.288807
C_1	6.575087	8.341685
C_2	6.586713	8.329319
C_a	6.674571	8.329319

Cuadro 1: Resultados modelos paramétricos

6. Resultados

6.1. Resultados anteriores

Encontramos resultados para comparar en el artículo Áccurate telemonitoring of Parkinson's disease progression by non-invasive speech tests'[2, pag 62-63]. Vemos que los resultados utilizando la técnica IRLS se sitúan con un error de test —utilizando la media del valor absoluto del error— de 6.71 para 'motor UPDRS'y 8.46 para 'total UPDRS'.

En el artículo considera también una técnica de basada en árboles llamada CART, con la que obtiene resultados de 5.77 para 'motor UPDRS'y 7.45 para 'total UPDRS'.

Además de estos resultados, que utilizan todas las variables de la base de datos, encontramos una lista de diferentes selecciones de características, con sus resultados.

6.2. Resultados de modelos paramétricos

En el Cuadro 1, podemos ver los resultados de los modelos paramétricos entrenados. Utilizamos, igual que en la sección anterior para poder comparar, el error de test utilizando la media del valor absoluto del error.

Los resultados obtenidos son similares a los modelos paramétricos del artículo. Encontramos también que tienen un comportamiento similar, donde tomar un conjunto de variables menor nos lleva a unos resultados ligeramente menores. Esto hace que la selección de uno u otro modelo, no afecte demasiado a la precisión final que obtenemos.

Sin embargo, hay una diferencia importante entre los modelos. Como bien sabemos, es preferible en igualdad de condiciones elegir el modelo más simple, la llamada navaja de Ockham, pues el modelo va a ser mejor a la hora de generalizarlo. Visto desde otra perspectiva, el modelo completo tiene una dimensión de Vapnik-Chervonenkis de 17 — 16 variables y el término independiente— mientras que, por ejemplo, C_2 tendrá solo una dimensión de 6.

Esto hace que la cota de generalización de C_2 sea considerablemente mejor. Por tanto, si tras realizar un estudio comparativo de ambas —mediante validación cruzada— la diferencia de precisión fuera pequeña, seleccionaríamos el modelo con menor número de características.

6.3. Resultados de modelo no paramétrico

Nuestro modelo de función de base radial, obtiene unos resultados de 5.01 para 'motor UPDRS'y 6.52 para 'total UPDRS'.

Los resultados nos muestran que tiene un resultado sensiblemente mejor que todos los aportados hasta el momento. Sin embargo, hay dos problemas de esta técnica, que hace que no sea aplicable a cualquier caso.

Por un lado, la ejecución de una predicción es muy lenta. La ejecución completa de la predicción del test para una de las variables a predecir, puede tardar varias horas. Es cierto que estando la técnica en funcionamiento para su uso real, lo más probable es que no sea necesario realizar tantas predicciones a la vez, pero es un factor a tener en cuenta para su aplicación.

Por otro lado, requiere una cantidad de datos muy grande para obtener resultados precisos. Este hecho, que es importante en el resto de modelos estudiado, es de vital importancia para estos tipos de modelos — función de base radial y K-NN—. En este caso en concreto, disponemos de muchas muestras de los pacientes para que funcione correctamente. Pero puede que al intentar aplicarse sobre otros datos, no sea así, y este modelo falle totalmente.

7. Bibliografía

Artículos consultados:

[1] Little MA, McSharry PE, Hunter EJ, Ramig LO (2009), 'Suitability of dysphonia measurements for telemonitoring of Parkinson's disease', IEEE Transactions on Biomedical Engineering, 56(4):1015-1022

[2] A Tsanas, MA Little, PE McSharry, LO Ramig (2009) Áccurate telemonitoring of Parkinson's disease progression by non-invasive speech tests', IEEE Transactions on Biomedical Engineering

Enlaces:

Base de datos: `https://archive.ics.uci.edu/ml/datasets/Parkinsons+Telemonitoring`

Función glm: `https://stat.ethz.ch/R-manual/R-devel/library/stats/html/glm.html`

Función LASSO: `https://cran.r-project.org/web/packages/glmnet/glmnet.pdf`