Revista Iberoamericana de Automática e Informática Industrial

Revista Iberoamericana de Automática e Informática Industrial 16 (2019) 369-379

www.revista-riai.org

SysGpr: Sistema de generación de señales sintéticas pseudo-realistas

F. León*, Fco. J. Rodríguez-Lozano, A. Cubero-Fernández, José M. Palomares, J. Olivares

Depto. Ingeniería Electrónica y Computadores. Edificio Leonardo da Vinci, Campus de Rabanales, Universidad de Córdoba, 14071 Córdoba, España.

Resumen

Las señales obtenidas desde sensores son ampliamente utilizadas en diferentes campos científicos. Sin embargo, no siempre se dispone de los recursos necesarios para obtener dichos datos, debido a limitaciones estructurales, físicas, económicas, ambientales, fallos en la recolección de los datos, etc. Es en este escenario limitante, donde se erige la generación de datos sintéticos. La generación de datos sintéticos tiene la característica de reducir tiempos de espera frente a los largos periodos temporales que necesitan algunos sensores para obtener grandes volúmenes de muestras. Además, los datos generados pueden llegar a ser todo lo robustos que los usuarios necesiten. Por ello este trabajo presenta un sistema de generación de señales sintéticas con carácter pseudo-realista para su uso aplicado a la validación de métodos y diseño de experimentos. El método de la generación de señales propuesto, hace uso de modelos estadísticos y el comportamiento del gradiente de la señal para ir generando nuevos datos. El sistema desarrollado se encuentra disponible públicamente como herramienta web.

Palabras Clave:
Análisis y tratamiento de señales, Diseño de experimentos, Modelado de señales, Datos sintéticos, Distribuciones estadísticas.

SysGpr: System of Generation of Pseudo-realistic Synthetic Signals.

Abstract

Signals obtained from sensors are widely used in different scientific fields. However, the resources to obtain the data are not always available due to structural constraints, physical, economic, environmental, and data collection failures, etc. It is in this scenario that the generation of synthetic data is established. The generation of synthetic data has several benefits, such as, reducing waiting times compared to the long periods required by some sensors to obtain large volumes of samples. In addition, the generated data can be as robust as users need it to be. For this reason, this paper presents a pseudo-realistic synthetic signal generation system for use in the validation of methods and design of experiments. The proposed signal generation method makes use of statistical models and the gradient of the signal to generate new data. The developed system is open for the public, available as a web tool.

Keywords:
Signal analysis and treatment, Design of experiments, Signal modelling, Synthetic data, Statistical distributions.

1. Introducción

En las última décadas se ha podido observar un gran avance de la tecnología y un aumento de la capacidad de cómputo así como la miniaturización de los sensores (Chen et al., 2016; Alee et al., 2016; Castrillón-Santan et al., 2014; Ollero et al., 2012). Todo esto ha dado como resultado el nacimiento del paradigma del Internet de las Cosas (IoT) (A. Biru and Rotondi, 2015). El gran auge del IoT ha hecho posible que hoy día se desplieguen redes de sensores donde antes era inviable (Kamila, 2017).

Sin embargo, el hecho de que hoy día se disponga de enormes cantidades de sensores y datos asociados a los mismos, no supone una ventaja en determinadas situaciones. En especial, los investigadores encuentran limitaciones a la hora de testear sus métodos debido a que en muchas ocasiones necesitan datos para validar sus experimentos. Y aunque se disponga de información procedente de sensores o redes de sensores, éstas

*Autor para correspondencia: fernando.leon@uco.es
To cite this article: F. León, Fco. J. Rodríguez-Lozano, A. Cubero-Fernández, José M. Palomares, J. Olivares. 2019. SysGpr: System of Generation of Pseudo-realistic Synthetic Signals. Revista Iberoamericana de Automática e Informática Industrial 16, 369-379. https://doi.org/10.4995/riai.2019.10025

pueden no ser accesibles por motivos de legalidad como ley de protección de datos al estar asociadas a usuarios, ser propiedad de empresas privadas, etc.

Además, obtener datos de una red de sensores lleva asociado un doble coste, debido a que desplegar una red de sensores puede ser costosa en términos económicos, y en términos temporales, dado que los sensores necesitan realizar un muestreo que en determinados casos conllevan una prolongación temporal de semanas o incluso meses, para poder disponer de datos y llevar a cabo una experimentación.

Como solución a los problemas comentados anteriormente, se dispone de diversos repositorios de datos tanto públicos como privados. Sin embargo, no siempre se adaptan a las necesidades del problema que se pretende tratar.

La solución a todas estas limitaciones es la generación de datos o señales sintéticas que emulen el comportamiento de la realidad del problema que se pretende abordar. Este tipo de señales tiene múltiples ventajas que ayudan a solventar las limitaciones citadas anteriormente (Hoag, 2008). Por un lado destaca la robustez, debido a que los sensores del mundo real que componen las redes de sensores, pueden proporcionar datos erróneos en determinados casos. Los datos obtenidos por un generador de datos sintéticos carecen de este problema.

Otra característica destacable es la seguridad, puesto que los datos sintéticos pueden generarse con un nivel de detalle y realismo tal, que no es necesario asumir ningún tipo de riesgo, frente a lo que puede suceder en algunas disciplinas de la ciencia como en medicina.

Los datos sintéticos como herramienta para poner a prueba métodos y modelos desarrollados se utilizan en diversos campos científicos, tales como reconocimiento y generación de patrones (Jiang et al., 2009), minería de datos (Peng and Hanke, 2016), en aprendizaje automático (Ekbatani et al., 2017), etc.

Este trabajo presenta un método para la generación de señales sintéticas basado en funciones de distribuciones estadísticas. Este trabajo se enmarca dentro del Proyecto ALCOR (F. Espinosa, 2018) para proporcionar conjuntos de datos con suficiente amplitud para poder realizar experimentaciones con un volumen elevado de datos. Al generar dichas señales de datos en base a distribuciones estadísticas se pueden conseguir datos que tengan un comportamiento similar al resultado que pueden proporcionar diversos sensores que existen en la actualidad tales como sensores con el fin de comprobar métodos y modelos como los trabajos realizados por (Dormido et al., 2008; Garcia-Alvarez and Fuente, 2011).

Una generación de señales de carácter pseudo-realistas, no puede consistir en obtener muestras aleatorias de distribución de probabilidad uniforme, pues el resultado sería una sucesión de valores carentes de toda coherencia. Debido a esto, se necesitan diferentes mecanismos para modificar la señal.

Junto con el modelo de generación de datos sintéticos con carácter pseudo-realistas que se propone en el presente trabajo, se proporciona una herramienta web disponible en (Leon et al., 2018) que implementa el modelo propuesto. Los datos generados pueden ser almacenados en el formato más usual de las bases de datos para experimentación (comma–separated values "CSV").

El presente documento se organiza de la siguiente forma: En la sección. 2 se muestran las propuestas de otros autores en la generación de señales sintéticas. La sección. 3 describe las restricciones de las señales que generará el modelo y el método de generación de señales propuesto. Los resultados y su análisis se muestran en la sección. 4. Finalmente, en la sección. 5 se muestran las conclusiones obtenidas de los experimentos realizados y en la sección. 6 se muestran las posibles mejoras aplicables al método desarrollado.

2. Trabajos relacionados

La generación de señales sintéticas ha sido utilizadas en muchos campos de la ciencia. Por ejemplo en (Kuchar, 2004), los autores proponen un modelo llamado *WGENK* para la generación de datos sintéticos orientados a agricultura. *WGENK* es una variación del modelo *WGEN* (CW and DA, 1984). En el trabajo de Kuchar et al. se generan datos tales como la radiación solar diaria, temperaturas mínimas y máximas precipitaciones, etc. Los autores logran alcanzar un modelo de generación que se aproxima a la realidad y realizan un contraste con datos reales para validar sus resultados.

En (Ayala-Rivera et al., 2013) los autores realizan una modificación de la herramienta *open-source Benerator* (Bergmann, 2013) y hacen uso de una base de datos que contiene el censo poblacional de Irlanda. Los autores logran demostrar en sus experimentos que haciendo uso de herramientas de generación de datos sintéticos, y con las restricciones adecuadas se pueden conseguir datos que contengan las mismas métricas estadísticas que los datos del mundo real.

Observando las propuestas de los diferentes autores, se observa que existe cierta tendencia en la literatura científica de usar lenguajes específicos para etiquetar los datos (Hoag, 2008). En (Hoag and Thompson, 2007) los autores proponen un método para la generación de grandes conjuntos de datos de forma paralela. Utilizan el lenguaje *SDDL* (*Synthetic Data Description Language*) dado que los datos tienen que ser generados con diferentes restricciones. Este lenguaje está basado en *XML*(*Extensible Markup Language*) y es utilizado por muchos generadores de datos sintéticos cuando se necesita etiquetar datos y agregar restricciones a la generación.

Otro ejemplo del uso de lenguajes de etiquetado, lo encontramos en (Anderson et al., 2014), donde los autores desarrollan un *framework* que hace uso de estructuras basadas en *XML* para generar grandes volúmenes de datos. El modelo de generación de los datos sintéticos propuestos por los autores se compone de dos fases. La primera es la generación de los ficheros *XML* junto con la extracción de características de los datos. Y la segunda fase es la generación de los datos basándose en diferentes distribuciones estadísticas tales como distribuciones de Poison, normales y geométricas. De los resultados experimentales los autores concluyen que los datos generados tienen un comportamiento similar a los datos reales bajo un coeficiente de confianza del 95 %.

En (Josh Eno, 2008) los autores realizan la generación sintética de datos haciendo uso del estándar abierto *PMML* (*Predictive Model Markup Language*) como puente entre la base de datos original y el fichero *SDDL* generado. Una vez han conseguido el fichero *SDDL* utilizan un método *PSDG* (*Parallel synthetic data generation*) para obtener el nuevo conjunto de datos. En la experimentación llevada a cabo demuestran que

utilizando la base de datos *Iris* (Fisher, 2011), los datos sintéticos comparten las mismas características que los datos originales y que no existen diferencias significativas entre ellos.

Donde ha tenido una gran acogida la generación de datos sintéticos ha sido en minería de datos, en reconocimiento de patrones y en aprendizaje automático. Por ejemplo en (Frasch et al., 2011) los autores utilizan la generación de datos sintéticos para validar métodos de aprendizaje automático y de minería de datos. En el trabajo de los autores se emplea un método denominado *WGKS* (*White Gaussians on k-simplex*), que genera datos mediante distribuciones gaussianas. Al estar enfocado a generación de datos para aprendizaje automático se controlan factores como el número de clases y el error bayesiano.

En (Peng and Hanke, 2016) los autores generan nuevos conjuntos de datos sintéticos por medio de árboles de decisión mediante una modificación del algoritmo *ID3* (*Iterative Dichotomiser 3*). Mediante el uso de los árboles de decisión los autores consiguen crear interdependencia entre los datos de los conjuntos de datos generados con la intención de obtener conjuntos de datos genéricos con los que testear cualquier aplicación de aprendizaje automático.

En métodos de minería de datos tales como agrupamiento y detección de *outliers* no existen en ocasiones conjuntos de datos que sean útiles para probar la eficacia de dichos métodos (Pei and Zaïane, 2006). Este es el hecho que motiva a los autores a crear un método que genera datos de forma sintética en base a diferentes distribuciones estadísticas, con un determinado número de *clusters*, un nivel de dificultad concreto y la capacidad de incorporar un determinado ruido en la generación, para simular aquellos patrones que serán *outliers*.

En el estudio de series espacio-temporales encontramos que en (Theodoridis et al., 1999), los autores proponen el uso del algoritmo *GSTD* (*Generate Spatio Temporal Data*). Este método ha sido desarrollado por los autores para la generación sintética de datos con carácter espacio-temporal en dos dimensiones. Dicho algoritmo es capaz de modificar los parámetros asociados a un objeto y modificar su posición y tamaño a lo largo de un determinado intervalo de tiempo. Los atributos asociados al objeto como el intervalo, pueden generarse mediante una función de probabilidad estadística normal o sesgada.

Con un objetivo similar al propuesto en el trabajo anterior, (Girod et al., 2004) proponen una generación de conjuntos de datos de series espacio-temporales centrada en la generación de datos sintéticos de topologías de redes de sensores irregulares. Mediante los experimentos realizados y los casos de estudio en los que se ha utilizando el sistema *DIMENSIONS* (Ganesan et al., 2003), los autores demuestran que los datos sintéticos poseen características similares a los datos reales.

Tras analizar los trabajos propuestos por otros autores, se detecta que en general no existe un sistema de generación de señales sintéticas de propósito general, sino que se centran en proporcionar soluciones para problemas específicos. Aunque en el presente trabajo se compartan aspectos en común con los trabajos analizados, como el uso de distribuciones estadísticas, la principal ventaja del método de generación propuesto es que consigue realizar la generación mediante el uso del gradiente de la señal y la combinación de diferentes niveles modificadores permitiendo controlar el comportamiento de la señal.

3. Método

El método propuesto utiliza funciones de generación de números aleatorios para construir una señal con un número determinado de muestras a partir de un rango acotado. Es necesario que el método de generación de señales sintéticas sea capaz de generar muestras digitales que sean verosímiles desde un punto de vista cualitativo. Además, las señales resultantes deben cumplir el teorema de (Nyquist, 1928) y (Shannon, 1949).

Como parámetros de entrada al modelo se consideran el número de muestras N, y el rango de valores posibles, representados por los valores frontera mínimo y máximo: s_m, s_M.

De acuerdo con (1) y (2), una señal de N muestras puede expresarse mediante la primera muestra y la señal correspondiente a los incrementos muestra a muestra de la señal original.

$$\forall\ s \in \mathrm{R}^n\ \exists\ s' \in \mathrm{R}^{n-1}\ /\ s'[i] = s[i+1] - s[i] \tag{1}$$

$$s[i] = s[0] + \sum_{j=0}^{i-1} s'[j] \tag{2}$$

Se introduce también un conjunto de distribuciones de probabilidad, cada una de ellas definida mediante un conjunto de parámetros reales cuya instanciación da lugar a una distribución de probabilidad concreta, Gamma y Gaussiana (Normal).

La distribución de probabilidad gamma consta de dos parámetros siempre positivos, α y β El primer parámetro, es el que representa la máxima intensidad de probabilidad y por tanto la forma de la distribución. Y el valor de β representa el alcance de la asimetría positiva hacia la derecha. Esta distribución dada sus características, permite moldear en determinados casos (Muñoz, 2014) otros tipos de distribuciones.

La distribución de probabilidad normal cuenta con los parámetros media (μ) y desviación típica (σ), y cada par de estos parámetros da lugar a una distribución de probabilidad diferente.

De manera general, la generación de N muestras acotadas entre s_m y s_M es un proceso que consta de las siguientes fases:

1. Se genera el valor inicial de la señal de manera aleatoria siguiendo una distribución de probabilidad uniforme.
2. Del conjunto de distribuciones de probabilidad, se escoge una aleatoriamente.
3. Cada parámetro que caracteriza la distribución de probabilidad resultante se genera aleatoriamente dentro de unos márgenes preconfigurados y, de nuevo, mediante una distribución de probabilidad uniforme.
4. Utilizando la distribución de probabilidad ya generada, se extraen N-1 muestras aleatorias, y se construye la señal con (2).

Con este método se obtiene una señal con coherencia (ya que sus incrementos siguen una distribución de probabilidad concreta) pero monótona. La naturaleza presenta variaciones que difícilmente se van a modelar satisfactoriamente utilizando una distribución de probabilidad constante. Para obtener señales cualitativamente más reales, la propuesta que se presenta es utilizar el procedimiento expuesto anteriormente para generar no solo variaciones en la señal, sino variaciones de los parámetros de la distribución de probabilidad que genera esta señal.

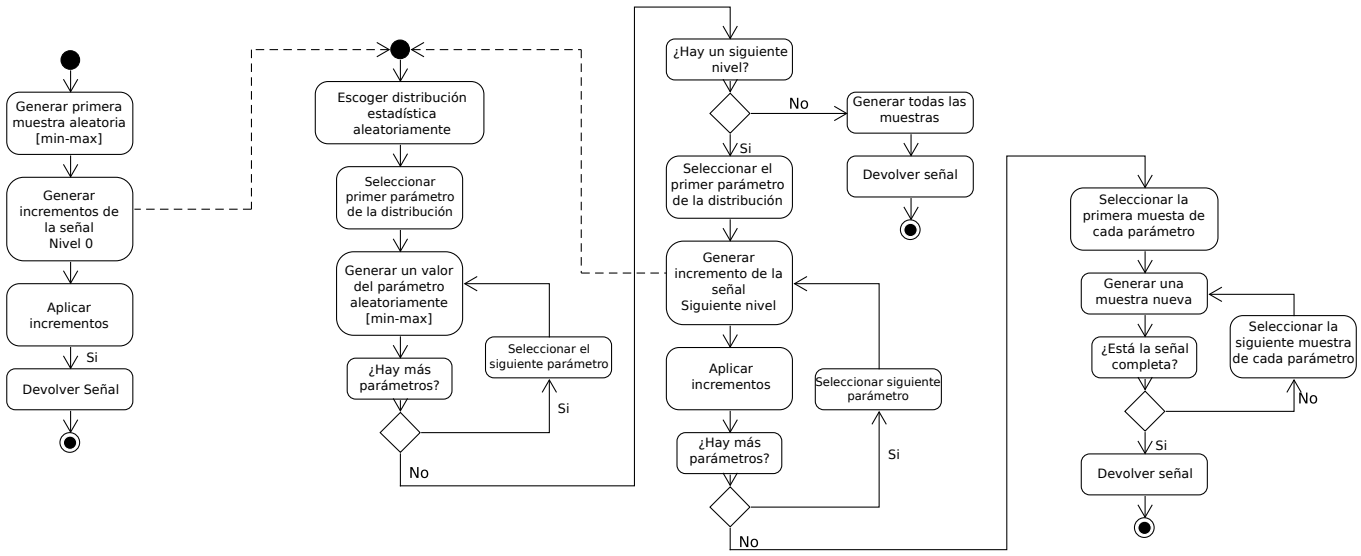

Figura 1: Diagrama de flujo del método propuesto

Este concepto da lugar a un procedimiento recursivo que se puede visualizar como un sistema de generación de señales por niveles, en el que el nivel 0 corresponde al nivel de la señal (el fin último del proceso), el nivel 1 corresponde a las distribuciones de probabilidad que generarán los incrementos de la señal, el nivel 2 corresponde a las distribuciones de probabilidad que generarán los incrementos que harán cambiar los parámetros de la distribución de probabilidad del nivel anterior, etc.

De manera genérica, el nivel de generación x es invocado para generar una señal aleatoria en el nivel anterior, para lo cual genera aleatoriamente la configuración inicial de una distribución de probabilidad escogida al azar y solicita al nivel de generación $x+1$ (si lo hubiere) que genere los cambios dinámicos de sus propios parámetros.

Un aspecto importante a tener en cuenta es la configuración necesaria a la hora de generar la señal. Como se ha mencionado anteriormente, cada tipo de distribución de probabilidad necesita unos parámetros de configuración que deben ser acotados uno a uno y cada nivel. Esta configuración permite controlar la aleatoriedad del comportamiento de la señal sin que se los sucesivos incrementos generados se descontrolen.

Por ilustrar este aspecto, considérese que se desea generar una señal cuyas muestras estén acotadas entre s_m y s_M; lo lógico es diseñar el primer nivel con unos parámetros de configuración para generar cambios que estén proporcionados con el margen dinámico $s_M - s_m$. Este procedimiento debe extrapolarse a todos los niveles, para que cada nivel aporte variaciones en una escala controlada al nivel anterior. Para una compresión más detallada sobre el funcionamiento del método, la Figura 1 muestra el diagrama de flujo del mismo. Además, en el Apéndice A se proporciona un ejemplo simplificado del código del método propuesto. Adicionalmente en el Apéndice B en las Figura B.8 y Figura B.9 se encuentran un diagrama de las clases de la implementación del método desarrollado y un diagrama de instancias que muestran un ejemplo de como son distribuidos los parámetros por las clases y las funciones utilizadas en el caso de dos niveles de generación.

4. Resultados experimentales

4.1. *Herramienta de generación de señales*

Con el fin de comprobar la utilidad del método, se ha realizado la implementación del mismo en una herramienta web llamada *"SysGpr"* la cual permite configurar y generar señales, disponible en (Leon et al., 2018).

En la Figura 2 se puede observar la pantalla principal de la aplicación. En la parte izquierda, la cabecera *"Configuration"* permite realizar la configuración básica de la señal modificando parámetros como el rango de valores y el número de muestras.

Las cabeceras *"Level 0* y *"Level 1"* permiten configurar los diferentes niveles (la barra de herramientas *"Levels"* permite añadir o eliminar niveles) de distribuciones estadísticas y habilitar o deshabilitar las distribuciones. En caso de tener seleccionadas dos distribuciones, para cada señal nueva que se genere se escoge una de las dos de forma aleatoria.

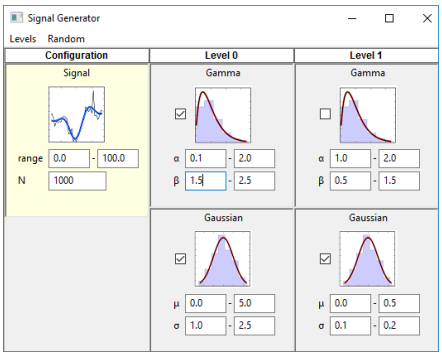

Figura 2: Interfaz para la generación de señales

Una vez establecidos los parámetros deseados para generar una señal, se puede tal y como se muestra en la Figura 3, generar tantas señales como se desee en una sola ejecución (esta

característica se encuentra dentro de *"Random"* en la barra de herramientas). Esta característica agiliza enormemente la tarea en caso de necesitar un gran número de señales para trabajar con ellas posteriormente.

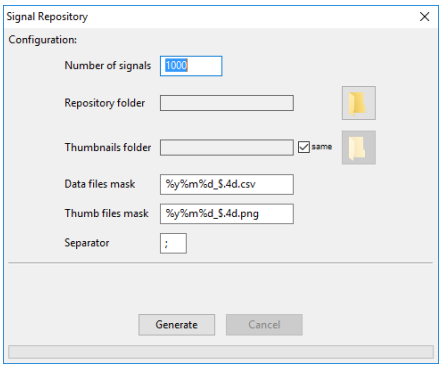

Figura 3: Generador de repositorios de señales

4.2. *Muestras generadas*

Para mostrar el funcionamiento del modelo, se han llevado a cabo cuatro generaciones de conjuntos de señales para que se puedan apreciar visualmente los resultados.

En el primer conjunto de señales sintéticas (T') se generan datos similares a los obtenidos por un sensor de temperatura *DHT22* (T). Los resultados de la generación de dichos datos se pueden observar en la Figura 4. La primera señal de esta figura se corresponde con los datos obtenidos desde el sensor de temperatura durante 200 min en un recinto climatizado y con fluctuaciones de aire que modificaban la temperatura. Los parámetros de generación para obtener datos similares a los obtenidos por dicho sensor, corresponden a un nivel de distribuciones gausianas con valores comprendidos entre 20°C y 22°C con un total de 200 muestras y $\mu = [0 - 0{,}1]$ y $\sigma = [0{,}01 - 0{,}1]$. Dichos valores han sido escogidos analizando la naturaleza de las señales de temperatura debido a que los cambios en temperatura en las condiciones del experimento son moderados.

El segundo conjunto (V') de experimentos se corresponde con la Figura 5. En dicha figura se representa en la gráfica de la izquierda la señal de velocidad obtenida por un anemómetro *NRG40* (V). En este caso las señales sintéticas se han generado con una función de distribución de nivel 0 gamma con un rango de valores entre 0 km/h y 1,5 km/h y un total de 200 muestras. Dado que la velocidad del viento puede cambiar drásticamente y tener un comportamiento muy pronunciado en algunos instantes la señal se ha generado con unos valores de $\alpha = [0{,}1 - 0{,}5]$ y $\beta = [0{,}1 - 0{,}5]$.

Para demostrar el uso de diferentes niveles de modificación en la Figura 6 se muestran el conjunto (P') generado con un nivel modificador de distribución gamma y un nivel inicial gausiano. Al igual que sucede con los casos anteriores, en la parte izquierda de la figura se encuentra la señal original, procedente de un sensor de presión atmosférica *BMP180* (P). El resto de figuras corresponden con algunas señales extraídas del conjunto de señales generadas con parámetros $\mu_{L_0} = [0{,}01 - 0{,}05]$, $\sigma_{L_0} = [0{,}01 - 0{,}03]$ y $\alpha_{L_1} = [0{,}01 - 0{,}03]$, $\beta_{L_1} = [0{,}01 - 0{,}05]$. La señal sintética tiene un rango de valores entre 1015 hPA y 1016 hPA y un total de 200 muestras. Estos parámetros han sido seleccionados debido a que la presión atmosférica raramente suele cambiar dependiendo del periodo de muestreo muy bruscamente. El segundo nivel añade un control de grado fino para ajustar el comportamiento de la señal.

La Figura 7 representa el último conjunto de señales sintéticas (I'), dedicado en esta ocasión a la generación de datos de irradiación solar. La gráfica de la izquierda corresponde con los datos reales obtenidos de un pirómetro *SP-215* (I) durante el mediodía en el mes de febrero con presencia de nubes en la azotea de uno de los edificios de la Universidad de Córdoba, España. Para modelar señales con un carácter realista que representen datos similares a los que proporciona el sensor, se ha propuesto utilizar dos niveles modificadores de la señal. Ambos niveles corresponden con señales gausiannas y tienen como valores de sus parámetros característicos: $\mu_{L_0} = [0 - 5]$, $\sigma_{L_0} = [1 - 2{,}5]$ y $\mu_{L_1} = [0 - 2]$, $\sigma_{L_1} = [0{,}5 - 1{,}5]$. El rango de valores de la señal está comprendido entre 200 w/m^2 y 300 w/m^2 con un total de 200 muestras.

4.3. *Validación*

La validación de los resultados del método propuesto se ha llevado a cabo mediante tres métricas diferentes.

4.3.1. *Validación mediante algoritmo de aprendizaje C4.5*

La primera validación se ha llevado a cabo utilizando el algoritmo de aprendizaje automático de árboles de decisión C4.5 (Quinlan, 1993). Se han realizado cuatro pruebas diferentes que corresponden a los diferentes conjuntos de datos generados en el apartado anterior (T', V', P', I'). Dado que se emplea un algoritmo de aprendizaje automático, es necesario entrenar el modelo para posteriormente llevar a cabo una validación o etapa de test. Los datos de entrenamiento para cada modelo incorporan 200 señales reales procedentes de cada sensor (T, V, P o I) y 200 señales procedentes de los otros sensores para que el modelo aprenda que tipo de señales no corresponden a los datos sensor que se pretende clasificar.

Para el conjunto de test se han utilizado 30 señales catalogadas como T, V, P e I, pero que realmente se tratan de señales pseudo-naturales de los conjuntos T', V', P' e I'. Por otro lado se han escogido otras 30 señales catalogadas como $\overline{T}$, $\overline{V}$, $\overline{P}$, $\overline{I}$ que también son señales pseudo-naturales que emulan el resto de sensores.

El motivo de utilizar 400 (200 + 200) y 60 (30 + 30) señales en las fases de entrenamiento y test, es que tal y como estipula el teorema central del límite, estas cifras se consideran suficientes y representativas de las poblaciones de datos de las que proceden.

De este modo, el primer modelo de clasificación dispondrá para su entrenamiento de un total de 400 señales. La mitad de ellas señales procedentes del sensor de temperatura (T) y el resto que forman el conjunto de entrenamiento etiquetadas como $\overline{T}$ son señales procedentes de V, P o I, que corresponden con los sensores restantes. En la fase de test, el conjunto estará formado por 30 señales de temperatura generadas con el método propuesto T' pero etiquetadas como T y 30 señales sintéticas de tipo V', P' o I' catalogadas como clase $\overline{T}$.

Hay que destacar que todas las señales, que se han utilizado para la fase de aprendizaje y Test, han sido normalizadas a valores en el rango $[0 - 1]$ para eliminar el efecto de sesgo aditivo

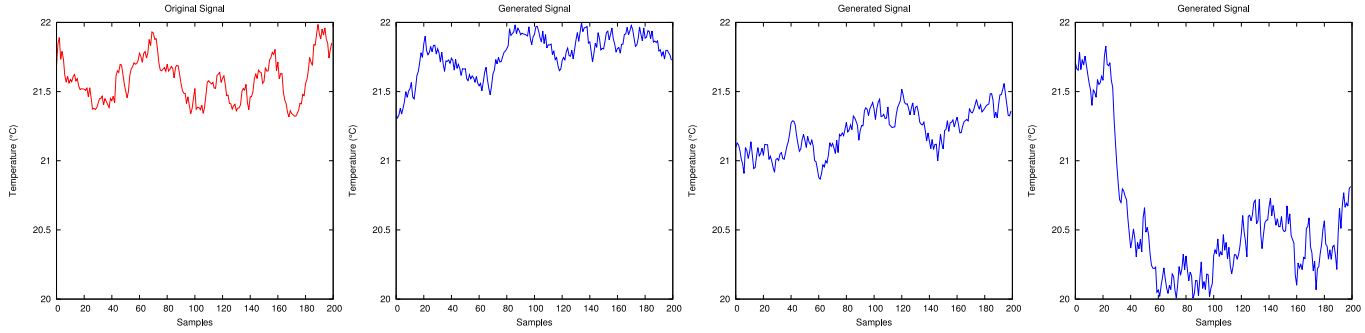

Figura 4: Señal original y señales generadas aleatoriamente utilizando solo el nivel 0 con distribución gaussiana

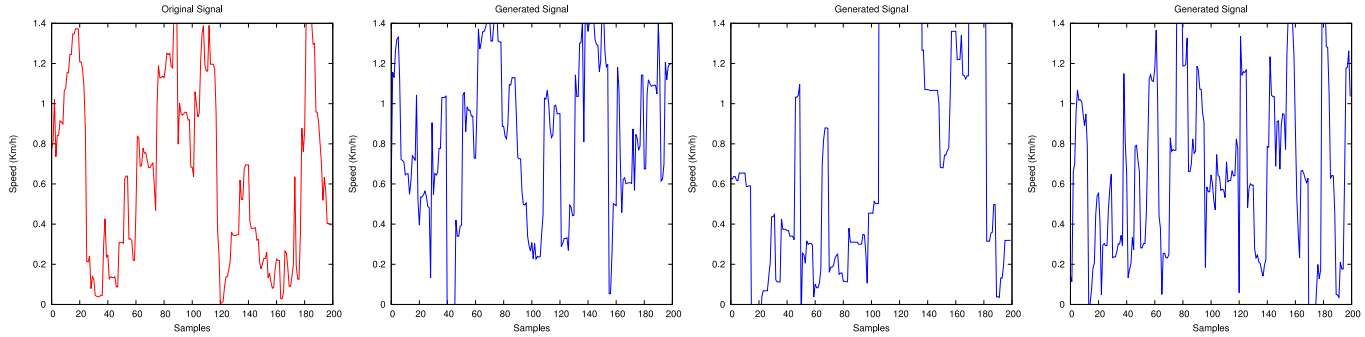

Figura 5: Señal original y señales generadas aleatoriamente utilizando solo el nivel 0 con distribución gamma

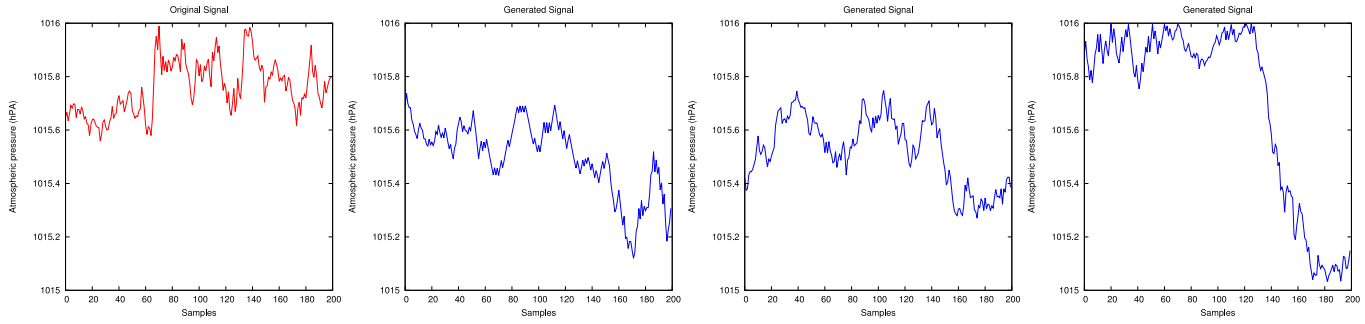

Figura 6: Señal original y señales generadas con distribución gaussiana de nivel 0 y nivel 1 de tipo gamma

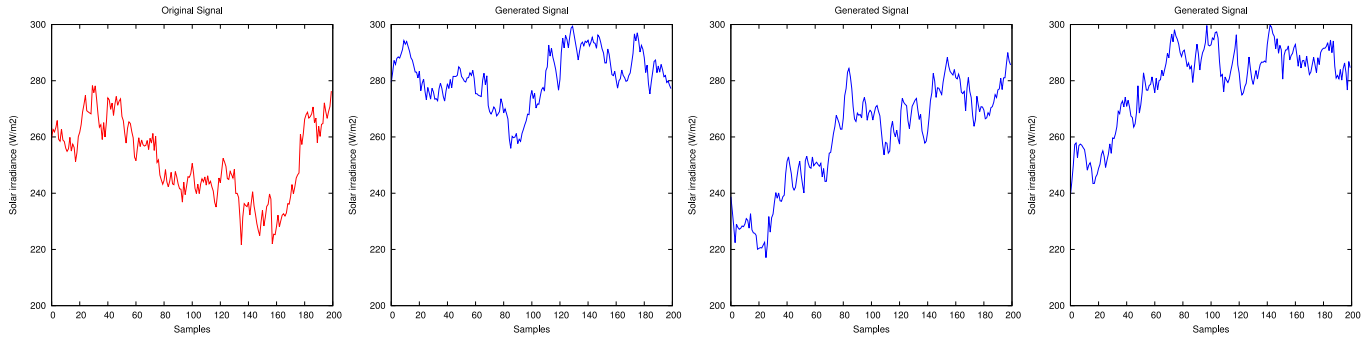

Figura 7: Señal original y señales generadas con distribuciones gaussiana en nivel 0 y en nivel 1

o cambios de escala, dado que el objetivo es comprobar que las señales tienen un comportamiento similar.

La siguiente lista detalla el tipo de señales que componen los conjuntos de entrenamiento y test para los diferentes modelos y casos estudiados:

- Modelo para señales de temperatura:
  - Datos para etapa de aprendizaje:
    - T: 200 señales obtenidas del sensor de temperatura *DHT22*.
    - $\overline{T}$: 200 señales procedentes de V, P e I.
  - Datos para etapa de Test:
    - T: 30 señales pseudo-naturales del conjunto T'.
    - $\overline{T}$: 30 señales pseudo-naturales procedentes de V', P' e I'.
- Modelo para señales de velocidad:
  - Datos para etapa de aprendizaje:
    - V: 200 señales obtenidas del anemómetro *NRG40*.
    - $\overline{V}$: 200 señales procedentes de T, P e I.
  - Datos para etapa de Test:
    - V: 30 señales sintéticas del conjunto V'.
    - $\overline{V}$: 30 señales pseudo-naturales procedentes de T', P' e I'.
- Modelo para señales de presión:
  - Datos para etapa de aprendizaje:
    - P: 200 señales obtenidas del sensor de presión atmosférica *BMP180*.
    - $\overline{P}$: 200 señales procedentes de T, V e I.
  - Datos para etapa de Test:
    - P: 30 señales pseudo-naturales del conjunto P'.
    - $\overline{P}$: 30 señales pseudo-naturales procedentes de T', V' e I'.
- Modelo para señales de irradiancia solar:
  - Datos para etapa de aprendizaje:
    - I: 200 señales obtenidas del pirómetro *SP-215*.
    - $\overline{I}$: 200 señales procedentes de T, V y P.
  - Datos para etapa de Test:
    - I: 30 señales pseudo-naturales del conjunto I'.
    - $\overline{I}$: 30 señales pseudo-naturales procedentes de T', V' y P'.

En la Tabla 1 se pueden observar los resultados del algoritmo de clasificación C4.5. En dicha tabla se muestra resumida toda la información de los datos obtenidos en la etapa de test del clasificador. En la parte de la izquierda se muestra la tabla de contingencia de cada uno de los modelos estudiados. Además, a la izquierda de cada tabla de contingencia, se proporcionan métricas como son el ratio de verdaderos positivos (TP) y Falsos Positivos (FP) y el área bajo la curva ROC.

Tabla 1: Resultado algoritmo clasificación C4.5 para las diferentes señales

	T	$\overline{T}$	TP	FP	ROC
T	22	8	0,733	0,167	0,783
$\overline{T}$	5	25	0,833	0,267	0,783
	V	$\overline{V}$			
V	22	8	0,733	0,367	0,693
$\overline{V}$	11	19	0,633	0,267	0,693
	P	$\overline{P}$			
P	25	5	0,833	0,233	0,789
$\overline{P}$	7	23	0,767	0,167	0,789
	I	$\overline{I}$			
I	23	7	0,767	0,233	0,802
$\overline{I}$	7	23	0,767	0,233	0,802

4.3.2. Validación mediante autocorrelaciones y correlaciones cruzadas

En segundo lugar se han realizado pruebas de autocorrelaciones cruzadas y correlaciones cruzadas entre las diferentes señales. Los resultados de las autocorrelaciones y correlaciones medias, pueden observarse en la Tabla 2.

Las autocorrelaciones cruzadas se han llevado a cabo tomando únicamente una muestra de 30 señales de un tipo en concreto de señales reales de temperatura (T), velocidad (V), presión (P) o Irradiancia (I), y se han calculado las correlaciones señal a señal con ellas mismas para comprobar el grado de similitud entre las señales originales.

Para las correlaciones cruzadas se han tomado un total de 30 muestras de señales aleatorias de cada conjunto T', V', P' o I', y se han comparado una a una con las señales escogidas para las autocorrelaciones de la misma naturaleza. Por ejemplo en el caso de las señales de temperatura se ha obtenido la correlación media entre T y T' para observar la similitud de las señales generadas con las señales reales.

En ambos casos se han escogido muestras aleatorias simples cumpliendo el teorema central del límite obteniendo así un número de muestras suficiente para que sean representativas de las poblaciones de los diferentes conjuntos. Además al igual que ocurría en la validación mediante el algoritmo de clasificación, las señales se han normalizado al rango $[0-1]$, para mitigar el efecto cambios de escala y poder comparar las señales en forma.

Puede observarse de los resultados obtenidos en la Tabla 2, que las señales pseudo-naturales guardan en media una similitud en torno al 80 % con las señales obtenidas de sensores reales, y que el 20 % se debe a la propia variación entre las señales originales. Debido a estas variaciones, en el caso de las señales pseudo-naturales de temperaturas se obtiene una correlación mayor que las propias autocorrelaciones cruzadas de las señales reales, lo que significa que hay más cantidad de señales generadas con una forma similar, existiendo una mayor homogeneidad que en las señales obtenidas de los sensores reales.

Tabla 2: Resultado de análisis de autocorrelaciones cruzadas y correlaciones cruzadas.

Tipo de señales	Autocorrelaciones	Correlaciones
Temperatura	T/T : 0,84056	T/T' : 0,84565
Velocidad	V/V : 0,81264	V/V' : 0,80436
Presión	P/P : 0,83740	P/P' : 0,83284
Irradiancia	I/I : 0,86898	I/I' : 0,85313

4.3.3. Validación mediante MOS (Mean Opinion Score)

Adicionalmente, se ha utilizado *Mean Opinion Score (MOS)* (ITU–T, 1996), para realizar una evaluación de calidad subjetiva de las señales pseudo-realistas con el fin de contrastar los resultados proporcionados por el generador de señales con señales reales proporcionadas por sensores.

Para llevar a cabo la validación mediante *MOS*, se han realizado una serie de encuestas a diferentes usuarios. Las encuestas pretenden evaluar de manera empírica la calidad de las señales generadas, éstas fueron realizadas por seis investigadores los cuales tienen conocimientos relacionados con el tratamiento de señales (el número de expertos escogidos, cumple los requisitos mínimos exigidos por MOS (Streijl et al., 2014) para obtener resultados concluyentes). En dichas encuestas, a los expertos se les pedía que clasificasen las diferentes señales que se les mostraban para comprobar si los científicos eran capaces de diferenciar las señales reales y las generadas por el método propuesto. A cada investigador se le mostraron dos bloques de noventa preguntas con cuatro imágenes de señales. Cada bloque correspondía con un tipo de señal generada: señales pseudo–naturales de un nivel modificador y señales pseudo–naturales de dos niveles modificadores. Dentro de las cuatro señales se encontraba una señal original y tres señales pseudo–naturales en orden aleatorio (dentro de cada uno de los grupos anteriormente descritos). A cada señal mostrada, cada experto debía etiquetarla como *"real"* o *"sintética"*, aunque podría dejarla sin responder en caso de no ser capaz de decidirse. Un ejemplo de los diferentes grupos de señales que analizaron los expertos, se pueden observar en las Figuras 4, 5, 6 y 7. Hay que notar, que en el presente documento las señales aparecen etiquetadas como *Original signal* (señales rojas) y *Generated signal* (señales azules), para que el lector pueda observar las diferencias entre señales generadas y originales. Sin embargo, los expertos no disponían de dicha información, ni de colores que las diferenciasen, para evitar dar información de la naturaleza de los datos que analizaban. Los únicos datos que los expertos conocían, eran la señal con la información de los ejes X e Y, que mostraba el tipo de señal, temperatura, velocidad del viento, presión atmosférica y radiación solar.

En la Tabla 3 se pueden observar los resultados obtenidos mediante *MOS*. Estos resultados han sido obtenidos comparando las respuestas de los diferentes expertos al clasificar los diferentes conjuntos de datos. Se han comprobado las etiquetas determinadas por los expertos, con las etiquetas reales que deben de tener cada señal pseudo–natural. El resultado fue que los expertos identificaron erróneamente o no pudieron distinguir en un 71,25 % de los casos si una señal procedía de una captura de datos real o de las proporcionadas por el generador de datos pseudo–naturales. Se observó que el 30 % de las señales pseudo–naturales de un único nivel modificador fueron clasificadas correctamente por los expertos. Asimismo, en el caso de las señales pseudo–naturales con dos niveles de modificadores, el 72,5 % señales pasaban desapercibidas a ojos de los expertos. En la Tabla 4 se pueden observar los resultados acumulados de las respuestas de la evaluación MOS. En dicha tabla, se pueden ver, por filas, el tipo de señales presentadas a los expertos, mientras que por columnas se puede ver la clasificación de cada señal por parte de los expertos. Se consideran aciertos aquellas señales reales identificadas como reales y aquellas señales sintéticas identificadas como sintéticas. Todas las demás combinaciones (incluidas todas aquellas señales que los expertos no sabían si eran reales o sintéticas y por tanto no contestaron) se consideran fallos.

Tabla 3: Tabla de resultados extraídos de MOS. Un nivel modificador (L1), Dos niveles modificadores (L2).

	Fallos y No Sabe		Aciertos	
Tipo de señales	Señales	%	Señales	%
L1	1512	70 %	648	30 %
L2	1566	72,5 %	594	27,5 %
Media (L1,L2)	71,25 %		28,75 %	

Tabla 4: Tablas de contingencia de MOS. Un nivel modificador (L1), Dos niveles modificadores (L2). En filas, tipo de señales presentadas a los expertos. En columnas, número de señales según la respuesta de los expertos para cada tipo.

Tipo de señales	Reales	Sintéticas	No sabe
L1			
Reales	48	66	426
Sintéticas	282	708	738
L2			
Reales	66	114	360
Sintéticas	306	528	786

5. Conclusiones

En el presente trabajo se ha propuesto un método para generar señales sintéticas que tengan cierta similitud a las que podrían obtenerse desde un sensor. Dicho método tiene la capacidad de generar datos en los que el comportamiento de la señal es capaz de seguir un comportamiento basado en distribuciones gaussianas y gamma, proporcionando un carácter realista.

El método propuesto aporta un enfoque distinto en el campo de generación de señales mediante la aleatorización y determinación del comportamiento del gradiente de la señal que se pretende generar. Esta característica permite añadir niveles para modificar la señal de forma que al añadir sucesivos niveles de modificación se puede controlar el comportamiento de la señal de una forma concreta para ajustarla a las características deseadas.

Además se ha desarrollado una interfaz visual que hace uso del método propuesto y proporciona a los científicos una manera simple de poder generar grandes repositorios de datos pseudo–naturales para que puedan testear sus propios métodos en los diferentes campos científicos, sin que tengan que esperar

para obtener los datos desde sensores reales o realizar etapas previas de tratamiento de datos para eliminar datos erróneos o incompletos.

Respecto a los resultados obtenidos se han realizado validaciones mediante tres enfoques distintos a diferentes conjunto de datos procedentes de sensores y a señales pseudo–naturales generadas con el método propuesto.

Para la primera validación se ha empleado el algoritmo de aprendizaje C4.5. Dicho algoritmo ha utilizado señales reales en la fase de entrenamiento, pero en la fase de test las señales han sido sustituidas por señales pseudo–naturales para comprobar la similitud de los datos generados. De los resultados se puede concluir que las señales sintéticas pseudo–naturales han sido clasificadas en media como señales reales en más de un 75 % (tomando los resultados de patrones bien clasificados) de los casos y que únicamente en aproximadamente un 25 % de los casos son detectadas como señales sintéticas. En cuando a las métricas para evaluar la calidad de los clasificadores se ha escogido el área bajo la curva ROC que en media es superior a 76 %.

Para la segunda validación se han calculado las correlaciones cruzadas para los cuatro conjuntos de datos detallados en la sección 4.2. Por un lado se han obtenido las autocorrelaciones cruzadas de los datos procedentes de los sensores reales obteniendo un 20 % de diferencia en media, que se debe principalmente a la variabilidad de las propias señales. Al calcular las correlaciones cruzadas entre los datos reales y los generados por el método propuesto, se observa que la relación en media entre ambas poblaciones es mayor al 80 %.

Por último se ha realizado una validación mediante *MOS* para determinar con ayuda de expertos en el campo de tratamiento de señales la eficacia del método propuesto. Dichos expertos han etiquetado los diferentes conjuntos de señales sin información previa que delatasen a las señales pseudo–naturales, como reales o sintéticas. De los resultados medios de dichas pruebas se ha obtenido que los expertos sólo son capaces de diferenciar las señales pseudo–naturales de las originales en un 28,75 % de los casos.

6. Trabajo futuro

Una de las principales mejoras consiste en añadir una variedad de distribuciones estadísticas nuevas a la aplicación y la posibilidad de crear señales con componentes periódicos, con lo que se ampliaría el abanico de posibilidades de la aplicación.

Por otro lado, la inclusión de métodos metaheurísticos y de aprendizaje automático permitiría obtener los parámetros estadísticos característicos de una señal o conjunto de señales dados, para generar nuevos conjuntos de datos sintéticos de forma automática, preservando la misma naturaleza de los datos de entrada (Alzantot et al., 2017).

Agradecimientos

Este trabajo ha sido parcialmente financiado mediante el proyecto DPI2013-47347-C2-2-R.

Referencias

A. Biru, R. M., Rotondi, D., 2015. Towards a definition of the internet of things (iot). Tech. rep., IEEE Tech. Rep.

Alee, N., Ehkan, P., Kamarudin, L. M., Harun, A., aug 2016. Size efficiency for sensor node with embedded processing unit. In: 2016 3rd International Conference on Electronic Design (ICED). IEEE.
DOI: 10.1109/iced.2016.7804642

Alzantot, M., Chakraborty, S., Srivastava, M., mar 2017. SenseGen: A deep learning architecture for synthetic sensor data generation. In: 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). IEEE.
DOI: 10.1109/percomw.2017.7917555

Anderson, J. W., Kennedy, K. E., Ngo, L. B., Luckow, A., Apon, A. W., oct 2014. Synthetic data generation for the internet of things. In: 2014 IEEE International Conference on Big Data (Big Data). IEEE.
DOI: 10.1109/bigdata.2014.7004228

Ayala-Rivera, V., McDonagh, P., Cerqueus, T., Murphy, L., 2013. Synthetic data generation using benerator tool. CoRR abs/1311.3312.

Bergmann, V., 2013. Data benerator tool. [OnLine] Available: http://databene.org/databene-benerator. [Accessed: 23-may-2017].

Castrillón-Santan, M., Lorenzo-Navarro, J., Hernández-Sosa, D., jul 2014. Conteo de personas con un sensor RGBD comercial. Revista Iberoamericana de Automática e Informática Industrial RIAI 11 (3), 348–357.
DOI: 10.1016/j.riai.2014.05.006

Chen, H., Xue, M., Mei, Z., Oetomo, S. B., Chen, W., dec 2016. A review of wearable sensor systems for monitoring body movements of neonates. Sensors 16 (12), 2134.
DOI: 10.3390/s16122134

CW, R., DA, W., 1984. Wgen: A model for generating daily weather variables. US Department of Agriculture, Agricultural Research Service, ARS-8.USDA, Washington, DC. 1984.

Dormido, S., Sánchez, J., Kofman, E., jan 2008. Muestreo, control y comunicación basados en eventos. Revista Iberoamericana de Automática e Informática Industrial RIAI 5 (1), 5–26.
DOI: 10.1016/s1697-7912(08)70120-1

Ekbatani, H. K., Pujol, O., Segui, S., 2017. Synthetic data generation for deep learning in counting pedestrians. In: Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods. pp. 318–323.
DOI: 10.5220/0006119203180323

F. Espinosa, J.L. Lázaro, J. O., 2018. Proyecto alcor: Contribuciones a la optimización del guiado remoto de robots en espacios inteligentes. Revista Iberoamericana de Automática e Informática industrial 15 (4), 416–426.
DOI: 10.4995/riai.2018.9199

Fisher, R. A., Jan. 2011. UCI Machine Learning Repository: Iris Data Set. http://archive.ics.uci.edu/ml/datasets/Iris.

Frasch, J. V., Lodwich, A., Shafait, F., Breuel, T. M., aug 2011. A bayes-true data generator for evaluation of supervised and unsupervised learning methods. Pattern Recognition Letters 32 (11), 1523–1531.
DOI: 10.1016/j.patrec.2011.04.010

Ganesan, D., Estrin, D., Heidemann, J., Jan. 2003. Dimensions: Why do we need a new data handling architecture for sensor networks? SIGCOMM Comput. Commun. Rev. 33 (1), 143–148.
DOI: 10.1145/774763.774786

Garcia-Alvarez, D., Fuente, M., jul 2011. Estudio comparativo de técnicas de detección de fallos basadas en el análisis de componentes principales (PCA). Revista Iberoamericana de Automática e Informática Industrial RIAI 8 (3), 182–195.
DOI: 10.1016/j.riai.2011.06.006

Girod, L., Govindan, R., Ganesan, D., Estrin, D., Yu, Y., aug 2004. Synthetic data generation to support irregular sampling in sensor networks. In: GeoSensor Networks. CRC Press, pp. 211–234.
DOI: 10.1201/9780203356869.ch12

Hoag, J. E., 2008. Synthetic data generation: Theory, techniques and applications. Ph.D. thesis, University of Arkansas, Fayetteville, AR, USA, aAI3317844.

Hoag, J. E., Thompson, C. W., mar 2007. A parallel general-purpose synthetic data generator. ACM SIGMOD Record 36 (1), 19–24.
DOI: 10.1145/1276301.1276305

ITU–T, 1996. Methods for subjective determination of transmissions quality. Recommendation P.800.

Jiang, F., Gao, W., Yao, H., Zhao, D., Chen, X., apr 2009. Synthetic data generation technique in signer-independent sign language recognition. Pattern Recognition Letters 30 (5), 513–524.
DOI: 10.1016/j.patrec.2008.12.007

Josh Eno, C. W. T., may 2008. Generating synthetic data to match data mining patterns. IEEE Internet Computing 12 (3), 78–82.
DOI: 10.1109/mic.2008.55

Kamila, N. K. (Ed.), 2017. Handbook of Research on Wireless Sensor Network Trends, Technologies, and Applications. IGI Global.
DOI: 10.4018/978-1-5225-0501-3

Kuchar, L., apr 2004. Using WGENK to generate synthetic daily weather data for modelling of agricultural processes. Mathematics and Computers in Simulation 65 (1-2), 69–75.
DOI: 10.1016/j.matcom.2003.09.009

Leon, F., Rodriguez-Lozano, F. J., Cubero-Fernandez, A., Palomares, J. M., Olivares, J., 2018. Sysgpr, servicio web para la generación de señales sintéticas. [OnLine] Available: https://www.uco.es/giia/sysgpr/. [Accessed: 23-Apr-2018].

Muñoz, F., aug 2014. Distribuciones poisson y gamma: Una discreta y continua relación. Prospectiva 12 (1), 99.
DOI: 10.15665/rp.v12i1.156

Nyquist, H., April 1928. Certain topics in telegraph transmission theory. Transactions of the American Institute of Electrical Engineers 47 (2), 617–644.
DOI: 10.1109/T-AIEE.1928.5055024

Ollero, A., Maza, I., Rodríguez-Castaño, A., de Dios, J. M., Caballero, F., Capitán, J., jan 2012. Proyecto AWARE. integración de vehículos aéreos no tripulados con redes inalámbricas de sensores y actuadores. Revista Iberoamericana de Automática e Informática Industrial RIAI 9 (1), 46–56.
DOI: 10.1016/j.riai.2011.11.007

Pei, Y., Zaïane, O., 2006. A synthetic data generator for clustering and outlier analysis. Tech. rep., Department of computing Science, University of Alberta.

Peng, T., Hanke, F., 2016. Towards a synthetic data generator for matching decision trees. In: Proceedings of the 18th International Conference on Enterprise Information Systems. SCITEPRESS - Science and and Technology Publications.
DOI: 10.5220/0005829001350141

Quinlan, R., 1993. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA.

Shannon, C. E., Jan 1949. Communication in the presence of noise. Proceedings of the IRE 37 (1), 10–21.
DOI: 10.1109/JRPROC.1949.232969

Streijl, R. C., Winkler, S., Hands, D. S., dec 2014. Mean opinion score (MOS) revisited: methods and applications, limitations and alternatives. Multimedia Systems 22 (2), 213–227.
DOI: 10.1007/s00530-014-0446-1

Theodoridis, Y., Silva, J. R. O., Nascimento, M. A., 1999. On the generation of spatiotemporal datasets. In: Advances in Spatial Databases. Springer Berlin Heidelberg, pp. 147–164.
DOI: 10.1007/3-540-48482-5_11

Apéndice A. Código simplificado del método propuesto

Apéndice A.1. Función generateSignal()

```
Generator::generateSignal()
Begin

  signal = []
  increments = first_level.
      generateIncrements(samples-1)

  previous_sample = random_number(max =
      maximum, min = minimum)
  signal.append(previous_sample)

  for inc in increments:
    sample = previous_sample +
        random_choice(-1,1)*inc
    if sample > maximum: sample -= 2*inc
    elif sample < minimum: sample += 2*
        inc
    signal.append(sample)
    previous_sample = sample
  return signal

End
```

Apéndice A.2. Función generateIncrements()

```
Generator_Level::generateIncrements(N)
Begin

 st_distribution = random_choice(
     GetDistributions())

 for parameter in st_distribution.
     GetParameters():
  parameter.SetRandomValue()

  signal = []
  if next_level == Null:
   while len(signal) < N:
    signal.append(st_distribution.
        generateSample())

   else:
    increments = []
    parameters = st_distribution.
        GetParameters()

     for i in len(parameters)
      increments.append(next_level.
          generateIncrements(N-1))
      j = 0
      signal.append(st_distribution.
          GenerateSample())

      while len(signal) < N:
       for i in len(parameters):
        parameter = parameters[i]
        inc = increments[i][j]
        previous_value = parameter.
            value
        parameter.value = previous
            previous + random_choice
            (-1,1)*inc
        if parameter.value > parameter.
            maximum: parameter.value -=
             2*inc
        elif parameter.value <
            parameter.minimum:
            parameter.value += 2*inc

        signal.append(st_distribution.
            GenerateSample())
        j++
      return signal

End
```

Apéndice B. Diagramas de clases e instancias

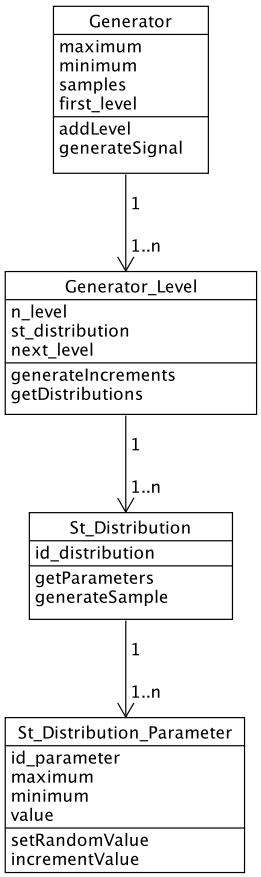

Figura B.8: Diagrama de clases.

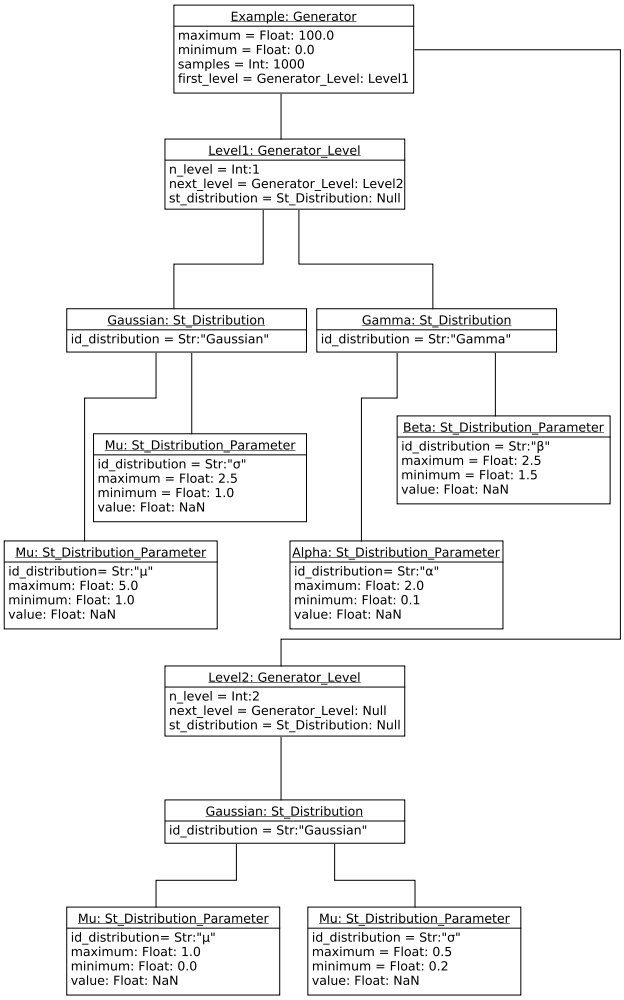

Figura B.9: Diagrama de instancias para dos niveles de generación.