

Análisis y modelado de series temporales con métodos estadísticos basados en Deep Learning

Doble Grado en Ingeniería Informática y Matemáticas

Miguel Lentisco Ballesteros

14 de septiembre de 2020

Trabajo Fin de Grado

E.T.S de Ingenierías Informática y de Telecomunicación
Facultad de Ciencias



**UNIVERSIDAD
DE GRANADA**

Índice de contenidos

Introducción y objetivos

Conceptos base

LSTM

Series temporales

Selección de modelos

Selección clásica

Perturbation Validation

Experimentación

Detección de anomalías

Detector

Alteraciones

Experimentación

INTRODUCCIÓN Y OBJETIVOS

Introducción

En los últimos años se ha dado una extensa popularización y uso de las **redes neuronales** que ha dado lugar a una gran cantidad de trabajos de investigación diversos.

Introducción

En los últimos años se ha dado una extensa popularización y uso de las **redes neuronales** que ha dado lugar a una gran cantidad de trabajos de investigación diversos.

De aquí surge nuestra para explorar en este campo ciertos temas que están **poco tratados** actualmente, en concreto, problemas con dominio de **series temporales** resueltos mediante redes neuronales **LSTM**.

Introducción

En los últimos años se ha dado una extensa popularización y uso de las **redes neuronales** que ha dado lugar a una gran cantidad de trabajos de investigación diversos.

De aquí surge nuestra para explorar en este campo ciertos temas que están **poco tratados** actualmente, en concreto, problemas con dominio de **series temporales** resueltos mediante redes neuronales **LSTM**.

Investigamos dos problemas de este tipo:

Introducción

En los últimos años se ha dado una extensa popularización y uso de las **redes neuronales** que ha dado lugar a una gran cantidad de trabajos de investigación diversos.

De aquí surge nuestra para explorar en este campo ciertos temas que están **poco tratados** actualmente, en concreto, problemas con dominio de **series temporales** resueltos mediante redes neuronales **LSTM**.

Investigamos dos problemas de este tipo:

- **Selección de modelos** para clasificación de series temporales.

Introducción

En los últimos años se ha dado una extensa popularización y uso de las **redes neuronales** que ha dado lugar a una gran cantidad de trabajos de investigación diversos.

De aquí surge nuestra para explorar en este campo ciertos temas que están **poco tratados** actualmente, en concreto, problemas con dominio de **series temporales** resueltos mediante redes neuronales **LSTM**.

Investigamos dos problemas de este tipo:

- **Selección de modelos** para clasificación de series temporales.
- **Detección de anomalías** en series temporales.

Objetivos

Los objetivos fundamentales de este proyecto son:

Objetivos

Los objetivos fundamentales de este proyecto son:

- Estudio y análisis de todo el **marco teórico** relacionado con el campo del aprendizaje profundo, con hincapié en la arquitectura **LSTM**; y las **series temporales**, incluyendo las técnicas y modelos más utilizados.

Los objetivos fundamentales de este proyecto son:

- Estudio y análisis de todo el **marco teórico** relacionado con el campo del aprendizaje profundo, con hincapié en la arquitectura **LSTM**; y las **series temporales**, incluyendo las técnicas y modelos más utilizados.
- Implementación y análisis del funcionamiento de una nueva heurística llamada *Perturbation Validation* utilizada para la **selección** de modelos.

Los objetivos fundamentales de este proyecto son:

- Estudio y análisis de todo el **marco teórico** relacionado con el campo del aprendizaje profundo, con hincapié en la arquitectura **LSTM**; y las **series temporales**, incluyendo las técnicas y modelos más utilizados.
- Implementación y análisis del funcionamiento de una nueva heurística llamada *Perturbation Validation* utilizada para la **selección** de modelos.
- Modelado y validación de un **detector** de series anómalas junto con el desarrollo de diversas **técnicas** que crean artificialmente series con perturbaciones.

CONCEPTOS BASE

CONCEPTOS BASE

LSTM

El **aprendizaje profundo** es el campo encargado de solucionar tareas del **aprendizaje automático** mediante el desarrollo de modelos basados en **redes neuronales**.

Modelo **bioinspirado** en el sistema neuronal del cerebro.

Redes neuronales

Modelo **bioinspirado** en el sistema neuronal del cerebro.

La versión más simple y conocida es la *feed-forward neural network* (FFNN).

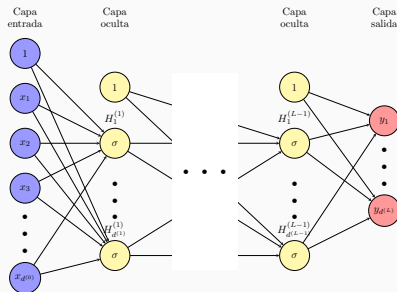


Figura 1: Estructura genérica de una *feed-forward neural network*.
Elaboración propia.

En el trabajo se ha realizado un desarrollo de los siguientes apartados:

- Orígenes.
- Explicación y funcionamiento de la FFNN.
- Tipología y arquitecturas actuales. **LSTM**.

LSTM

La arquitectura LSTM (*Long Short Term Memory*) es un tipo de red neuronal **recurrente** especializado para **datos secuenciales**, ya que de cada entrada se devuelve una salida y además cierta información que vuelve a la propia red, **retroalimentándose**, usada para la siguiente entrada.

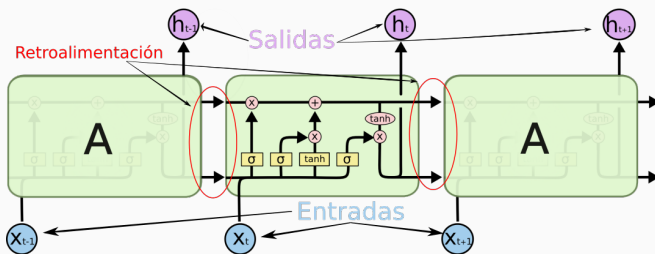


Figura 2: Esquema de funcionamiento de las neuronas LSTM. Extraído de [enlace].

CONCEPTOS BASE

Series temporales

Definición (Proceso estocástico)

Sean un espacio probabilístico (Ω, \mathcal{A}, P) , un espacio Borel (E, \mathcal{B}_E) , un conjunto ordenado arbitrario T y las variables aleatorias $X_t : (\Omega, \mathcal{A}, P) \rightarrow (E, \mathcal{B}_E)$, $\forall t \in T$. Se dice que la familia de variables aleatorias ordenadas por T , $\{X_t\}_{t \in T}$, es un **proceso estocástico**.

Definición (Proceso estocástico)

Sean un espacio probabilístico (Ω, \mathcal{A}, P) , un espacio Borel (E, \mathcal{B}_E) , un conjunto ordenado arbitrario T y las variables aleatorias $X_t : (\Omega, \mathcal{A}, P) \rightarrow (E, \mathcal{B}_E)$, $\forall t \in T$. Se dice que la familia de variables aleatorias ordenadas por T , $\{X_t\}_{t \in T}$, es un **proceso estocástico**.

Definición (Serie temporal)

Sea un proceso estocástico $\{X_t\}_{t \in T}$, una **serie temporal** es una realización muestral del proceso, denotado por $\{x_t\}_{t \in T}$.

Se han desarrollado las siguientes partes:

- Teoría probabilística y de procesos estocásticos.
- Modelos estacionarios.
- **Descomposición** y diferenciación de series.
- Discretización de series.

Descomposición de series

Una técnica muy útil para el análisis de series es la **descomposición** en tres componentes esenciales:

Descomposición de series

Una técnica muy útil para el análisis de series es la **descomposición** en tres componentes esenciales:

- **Tendencia:** cambio en el nivel medio de la serie a largo plazo.

Descomposición de series

Una técnica muy útil para el análisis de series es la **descomposición** en tres componentes esenciales:

- **Tendencia:** cambio en el nivel medio de la serie a largo plazo.
- **Estacionalidad:** patrón que se repite cada cierto periodo de tiempo.

Descomposición de series

Una técnica muy útil para el análisis de series es la **descomposición** en tres componentes esenciales:

- **Tendencia:** cambio en el nivel medio de la serie a largo plazo.
- **Estacionalidad:** patrón que se repite cada cierto periodo de tiempo.
- **Residuos:** valores sobrantes.

Descomposición de series

Una técnica muy útil para el análisis de series es la **descomposición** en tres componentes esenciales:

- **Tendencia:** cambio en el nivel medio de la serie a largo plazo.
- **Estacionalidad:** patrón que se repite cada cierto periodo de tiempo.
- **Residuos:** valores sobrantes.

La descomposición se suele denotar como:

$$x_t = m_t + s_t + Y_t, \forall t \in T,$$

donde m_t es la tendencia, s_t es la estacionalidad e Y_t los residuos.

Descomposición STL

Una de las formas más usadas es la **descomposición STL** que solo necesita el periodo de la componente estacional.

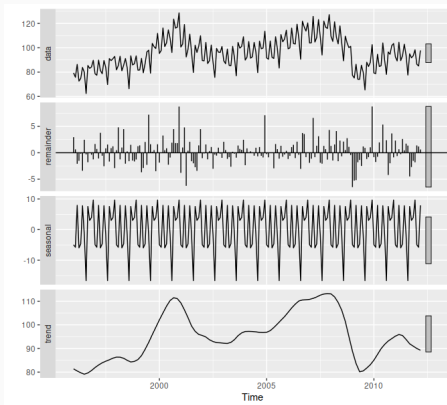


Figura 3: Ejemplo de descomposición STL. Extraído de [3].

SELECCIÓN DE MODELOS

SELECCIÓN DE MODELOS

Selección clásica

Pregunta: tengo varios modelos que han aprendido para resolver cierta tarea de clasificación de series temporales, ¿cual es el mejor?

Pregunta: tengo varios modelos que han aprendido para resolver cierta tarea de clasificación de series temporales, ¿cual es el mejor?

Respuesta: el que tenga mejor acierto (*acc*).

Pregunta: tengo varios modelos que han aprendido para resolver cierta tarea de clasificación de series temporales, ¿cual es el mejor?

Respuesta: el que tenga mejor acierto (*acc*).

Problema: sobreajuste.

Selección clásica

Solución: dividir los datos en dos conjuntos: uno de entrenamiento y otro de **validación**. Seleccionamos en base al *acc* en el conjunto de validación.

Selección clásica

Solución: dividir los datos en dos conjuntos: uno de entrenamiento y otro de **validación**. Seleccionamos en base al **acc** en el conjunto de validación.

Se suele usar la **validación cruzada** para tener una mejor estimación.

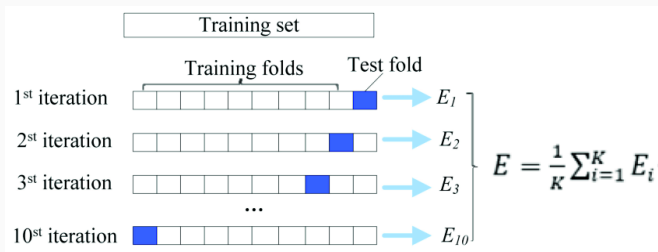


Figura 4: Esquema de validación cruzada. Extraído de [enlace].

Inconvenientes:

- Datos de validación insuficientes para representar la distribución subyacente.
- Sesgo en la muestra.
- No tiene en cuenta la complejidad.

SELECCIÓN DE MODELOS

Perturbation Validation

Perturbation Validation (PV) es una heurística que evalúa el ajuste del modelo frente a su espacio de hipótesis, midiendo el nivel de cambio del *acc* obtenido en varios conjuntos de etiquetas perturbadas.

Perturbation Validation (PV) es una heurística que evalúa el ajuste del modelo frente a su espacio de hipótesis, midiendo el nivel de cambio del *acc* obtenido en varios conjuntos de etiquetas perturbadas.

Si el modelo ha aprendido correctamente el patrón subyacente en los datos, no sobreajustará las etiquetas incorrectas.

Perturbated Validation

Perturbation Validation (PV) es una heurística que evalúa el ajuste del modelo frente a su espacio de hipótesis, midiendo el nivel de cambio del *acc* obtenido en varios conjuntos de etiquetas perturbadas.

Si el modelo ha aprendido correctamente el patrón subyacente en los datos, no sobreajustará las etiquetas incorrectas.

Tiene en cuenta el ajuste y la complejidad del modelo sin tener que realizar una partición de los datos.

1. Se crean k conjuntos de etiquetas perturbadas en base a unos ratios de error r_i , $i = 1, \dots, k$.

1. Se crean k conjuntos de etiquetas perturbadas en base a unos ratios de error r_i , $i = 1, \dots, k$.
2. Para cada conjunto de etiquetas, el modelo se entrena desde cero con dicho conjunto y se evalúa el ajuste con los mismos datos, obteniendo acc_i , $i = 1, \dots, k$.

1. Se crean k conjuntos de etiquetas perturbadas en base a unos ratios de error r_i , $i = 1, \dots, k$.
2. Para cada conjunto de etiquetas, el modelo se entrena desde cero con dicho conjunto y se evalúa el ajuste con los mismos datos, obteniendo acc_i , $i = 1, \dots, k$.
3. PV es la pendiente absoluta de la regresión lineal obtenida con los puntos $\{(r_i, acc_i)\}_{i=1}^k$.

Funcionamiento

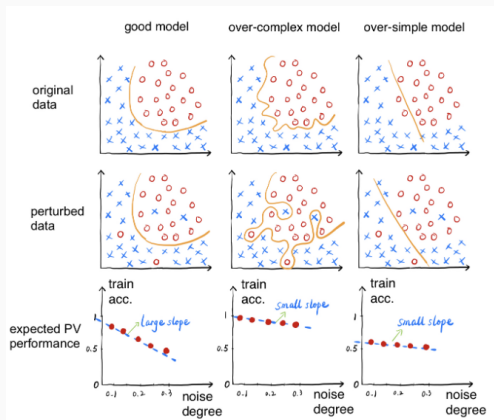


Figura 5: Idea de funcionamiento del PV. Extraído de [4]

SELECCIÓN DE MODELOS

Experimentación

Realizamos el siguiente experimento para medir la eficacia del *PV*:

- 114 *datasets* de series temporales unidimensionales de tareas de clasificación, divididos en entrenamiento y test.
- 11 modelos de clasificación usuales: nuestro modelo LSTM, SVM, k -NN y distintos árboles.
- Métricas medidas: *PV*, acc_{5CV} , acc_{train} , acc_{test} .

Resultados

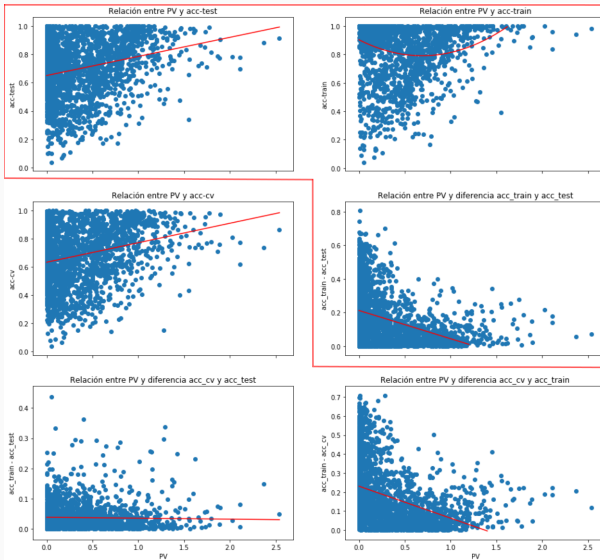


Figura 6: Resultados del experimento. Elaboración propia.

Concluimos que:

- $PV \uparrow \implies E[acc_{train}] \uparrow, E[acc_{test}] \uparrow$.
- PV es inversamente proporcional al sobreajuste.

Concluimos que:

- $PV \uparrow \implies E[acc_{train}] \uparrow, E[acc_{test}] \uparrow$.
- PV es inversamente proporcional al sobreajuste.

Pero...

- $PV(clf_1) > PV(clf_2) \not\Rightarrow acc_{test}(clf_1) > acc_{test}(clf_2)$.
- Los modelos con alta varianza (LSTM) producen valores atípicos.

PV no **sustituye** a CV pero puede **complementarlo**:

1. Seleccionamos los modelos con CV más altos.
2. Calculamos el PV para estos modelos.
3. Escogemos el modelo con PV más alto si $PV \gg 0$.

Hiperparámetros

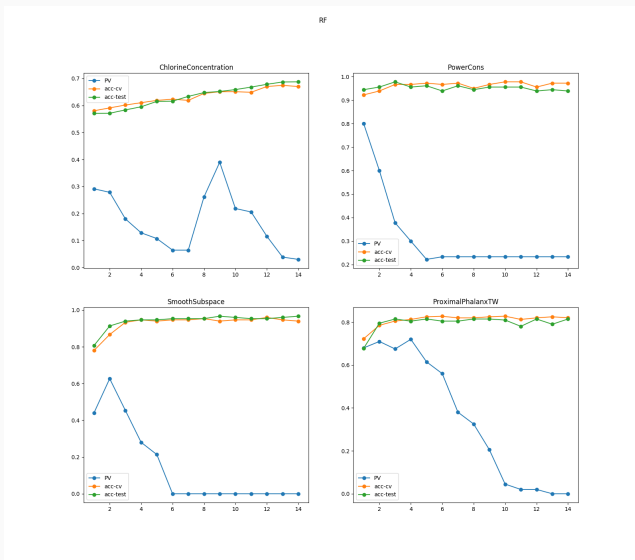


Figura 7: Experimento con hiperparámetros. Elaboración propia.

DETECCIÓN DE ANOMALÍAS

Una serie temporal es **anómala** cuando se producen valores inesperados en ciertos instantes.

Una serie temporal es **anómala** cuando se producen valores inesperados en ciertos instantes. Aunque dependen de la tarea concreta existen ciertos tipos usuales: cambios de intensidad puntuales o en un periodo, cambio de forma...

Una serie temporal es **anómala** cuando se producen valores inesperados en ciertos instantes. Aunque dependen de la tarea concreta existen ciertos tipos usuales: cambios de intensidad puntuales o en un periodo, cambio de forma...

Abordamos dos problemas en este campo:

- Falta de detectores desplegables en **diversas** tareas.
- Falta de *datasets* con muestras de series anómalas.

DETECCIÓN DE ANOMALÍAS

Detector

Modelamos un detector de series anómalas extensible para muchos casos, estructurado en dos partes:

- *Autoencoder* LSTM para reconstrucción.
- Predictor basado en la distribución de los errores.

Autoencoder LSTM

Un *autoencoder* LSTM que aprende el **patrón** de las series normales. Reconstruye las entradas para el patrón aprendido.

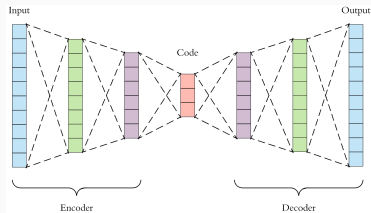


Figura 8: Arquitectura autoencoder. Extraído de [enlace].

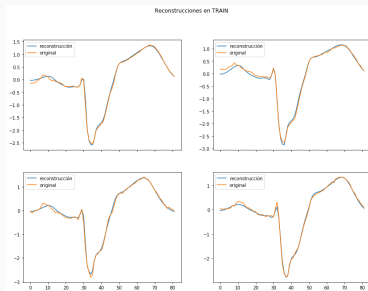


Figura 9: Ejemplos de reconstrucciones. Elaboración propia.

Probabilidad de anomalía asignada en base a la **distribución** aprendida de los errores cuadráticos medios entre las series normales y las reconstrucciones.

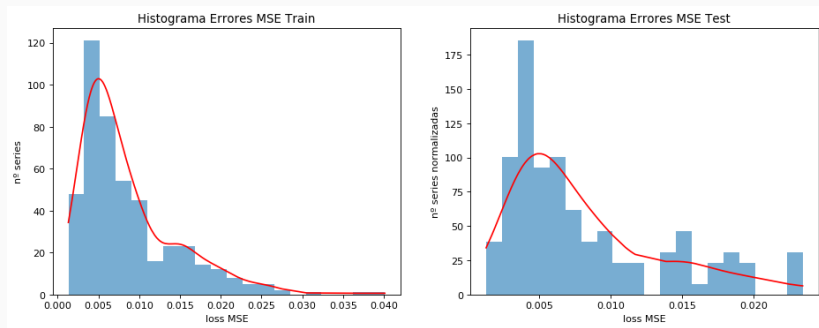


Figura 10: Ejemplo de distribución de errores. Elaboración propia.

DETECCIÓN DE ANOMALÍAS

Alteraciones

Cuatro métodos para alterar series normales que permiten:

- Validar detectores de anomalías cuando solo hay series normales.
- Crear muestras anómalas de entrenamiento para clasificadores.

Cuatro métodos para alterar series normales que permiten:

- Validar detectores de anomalías cuando solo hay series normales.
- Crear muestras anómalas de entrenamiento para clasificadores.

Siempre se selecciona un tramo aleatorio de la serie y se modifica.

Ruido gaussiano

Simulamos cambios grandes puntuales en la intensidad de la serie (picos) mediante **distribución gaussiana**.

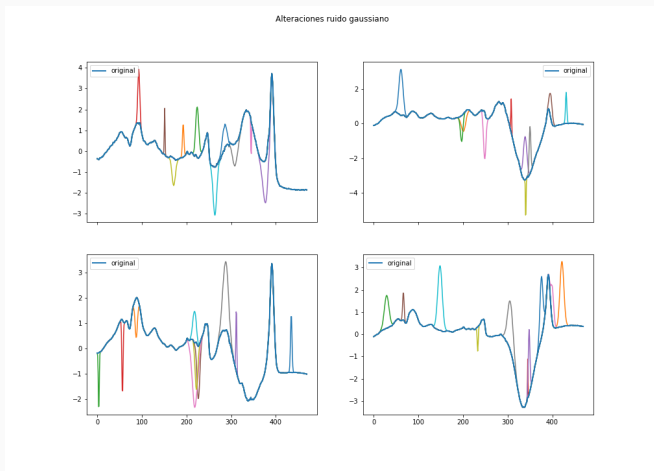


Figura 11: Perturbaciones con ruido gaussiano. Elaboración propia.

Pulso gaussiano-sinusoidal

Simulamos cambios de forma o interferencias en las series mediante un **pulso gaussiano-sinusoidal**.

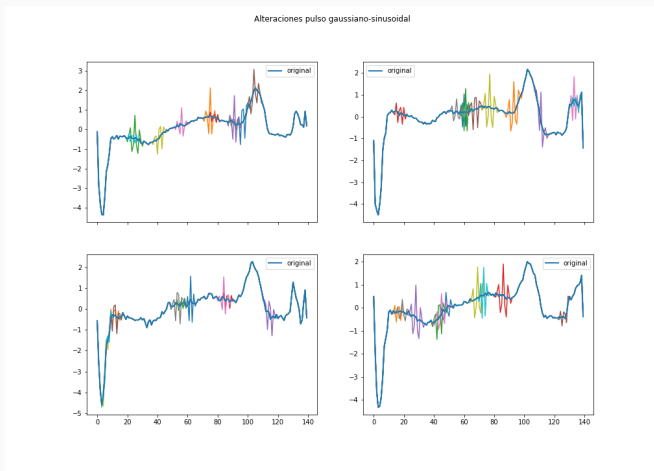


Figura 12: Alteraciones pulso g.s. Elaboración propia.

Intensidad estacionalidad

Simulamos cambios estacionales alterando la intensidad de la estacionalidad de la serie.

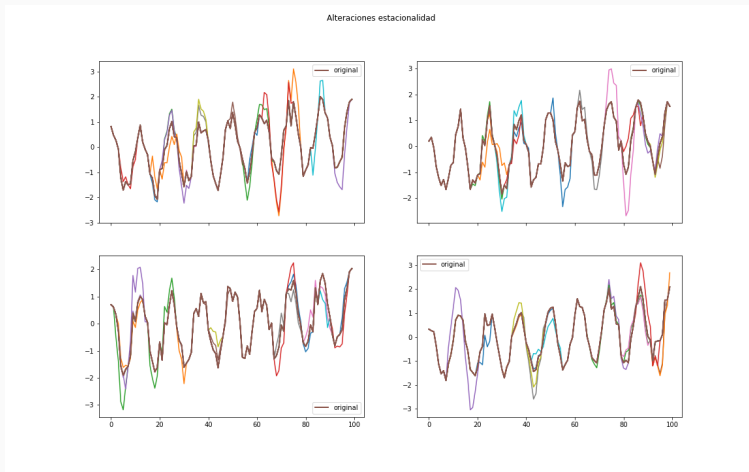


Figura 13: Perturbaciones con estacionalidad. Elaboración propia.

Intensidad tendencia

Simulamos cambios estacionales alterando la intensidad de la **tendencia** de la serie.

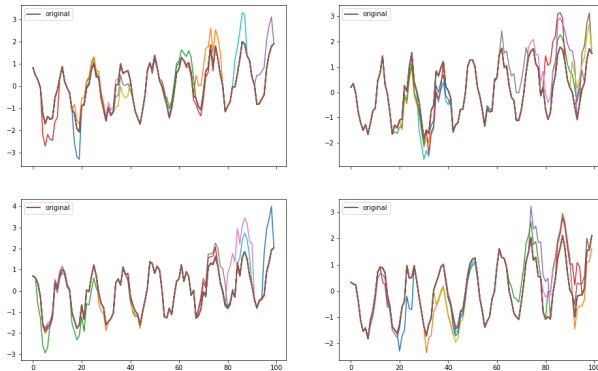


Figura 14: Perturbaciones con tendencia. Elaboración propia.

DETECCIÓN DE ANOMALÍAS

Experimentación

Validamos empíricamente la eficacia del detector para cada método de alteración propuesto en un par de *datasets* sin muestras anómalas.

Validamos empíricamente la eficacia del detector para cada método de alteración propuesto en un par de *datasets* sin muestras anómalas.

Probamos con distintos parámetros para crear las series anómalas y observamos el cambio de comportamiento del detector frente a los distintos valores tomados.

Resultados

En general hemos observado que:

- El detector consigue unos buenos resultados de la métrica.
- Las alteraciones se comportan adecuadamente.

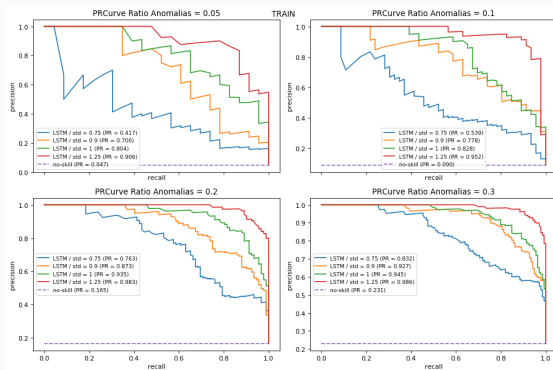


Figura 15: Ejemplo de validación del detector. Elaboración propia.

En este trabajo hemos logrado:






En este trabajo hemos logrado:

- Estudiar y comprobar la efectividad de *Perturbation Validation*, que puede complementar el método clásico de selección de modelos.

En este trabajo hemos logrado:

- Estudiar y comprobar la efectividad de *Perturbation Validation*, que puede complementar el método clásico de selección de modelos.
- Modelar un detector de series anómalas efectivo y fácil de desplegar en diversas tareas, junto con métodos para crear muestras anómalas para su uso en validación o entrenamiento.

Referencias

-  ABU-MOSTAFA, Y. S., MAGDON-ISMAIL, M. & LIN, H. T. (2012). *Learning from data*.
-  GOODFELLOW, I., BENGIO, Y. & COURVILLE, A. (2016). *Deep Learning*.
-  HYNDMAN, R. J. & ATHANASOPOULOS, G. (2018). *Forecasting: principles and practice*.
-  ZHANG, J. M., HARMAN, M., GUEDJ, B., BARR, E. T. & SHAW-TAYLOR, J. (2019). *Perturbation validation: a new heuristic to validate machine learning models*.
-  AHMED, M., MAHMOOD, A. N. & HU, J. (2016). *A survey of network anomaly detection techniques*.

Gracias por su atención.