Predicción a corto plazo de radiación solar
Short term solar radiation prediction

Trabajo de Fin de Máster
Curso 2018–2019

Autor
Nicolás Bueno Mora

Director
José Ignacio Gómez Pérez

Máster en Internet de las Cosas
Facultad de Informática
Universidad Complutense de Madrid

Predicción a corto plazo de radiación solar
Short term solar radiation prediction

Trabajo de Fin de Máster en Internet de las Cosas
Departamento de Arquitectura de Computadores y Automática

Autor
Nicolás Bueno Mora

Director
José Ignacio Gómez Pérez

Convocatoria: *Septiembre 2019*
Calificación: *6*

Máster en Internet de las Cosas
Facultad de Informática
Universidad Complutense de Madrid

Agradecimientos

Especial mención a mi director por la paciencia y dedicación para llevar este trabajo a buen puerto.

Agradecer a todos los que se han ido interesando sobre mi progreso durante la elaboración de este estudio.

Resumen

La predicción de los niveles de irradiación solar a corto plazo es relevante para obtener la energía que se va a producir y hacer el mejor uso. Existen modelos capaces de proporcionar con cierto grado de satisfacción esta predicción. Además de ser precisa, conviene que sea eficiente en recursos, reduciendo los requisitos al máximo posible del aprendizaje máquina. El enfoque de este trabajo es una exploración para hacer modelos más eficientes, intentando mejorar la predicción si es posible haciendo uso de modelos especializados.

Los datos de irradiación con los que se trabaja provienen de la red ubicada en Oahu, Hawai con diecisiete estaciones de medida durante un periodo de diecinueve meses con una frecuencia de un segundo. Este trabajo busca modelos creados con una red neuronal a partir de subconjuntos de todos estos datos mediante la aplicación de distintos métodos de análisis de afinidad entre los valores de irradiación, considerando por un lado la elección manual de los modelos especializados y por otro lado una selección de subconjuntos de datos automática, reduciendo así el impacto en los resultados finales de este trabajo de como son elegidos los modelos .

Los resultados de los modelos especializados elegidos con un conjunto representativo de datos no mejoran la calidad de la predicción pero se pueden tener en cuenta por conseguir resultados similares con menos recursos. Además de la obtención una mayor cantidad de datos correspondiendo a cada modelo, hacer uso de otros criterios para seleccionar modelos especializados serían líneas de posible mejora en la predicción.

Palabras clave

Internet de las Cosas, irradiación solar, redes neuronales, Tensorflow, Keras, correlación cruzada, PCA, aprendizaje máquina

Abstract

Predicting solar irradiance levels in the short term is important to have an accurate idea of the energy to be obtained and make the best use of it. There are models able to achieve this result to a certain extent. Is important to have both an accurate and efficient model, reducing as much as possible its machine learning requirements. The idea of this work is finding more efficient models, even more accurate by defining specialised models.

The data used here comes from an Oahu, Hawai based network consisting of seventeen measuring stations registering each one of them the solar irradiance at one second intervals. This work creates models from a neural network after selecting subsets of the data according to the results of applying some affinity analysis on the data. Selecting one where the final definition of the specialised model is made manually based on the aforementioned analysis, and one where the different models are automatically created by the irradiation values, the fact of how the models are chosen has less impact on the final result of the study.

Even if the results of the selected specialised models do not improve the quality of the prediction compared to the original generic model, those having a sufficient amount of data are highly recommandable to use because they perform almost as well as the original model with less resources. In addition to increasing the amount of data for each model, studying other criteria to define models, particularly when is manually, might be ways to improve solar prediction.

Keywords

Internet of Things, solar irradiance, neural networks, Tensorflow, Keras, cross correlation, PCA, machine learning

Índice

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

1.1. Motivación

La energía es una entidad esencial para las actividades humanas. Habiendo explorado distintas opciones a lo largo de la historia como los combustibles fósiles y en base tanto a los efectos destructivos sobre el entorno así como a su presencia finita y bastante limitada, cada vez cobra más importancia el buscar y optimizar fuentes alternativas más respetuosas con el medio ambiente y con mayor disponibilidad. Estas fuentes se agrupan bajo la categoría de energías renovables como pueden ser la mareomotriz (a partir de las olas del mar), eólica (aprovechando los vientos en superficie) o la solar, portando este trabajo sobre la radiación solar.

La radiación solar se transforma en energía eléctrica a través de las placas solares. La irradiación es por naturaleza variable en intensidad, sin ir más lejos en un punto fijo con el paso del día no se recibe la misma cantidad al amanecer que al medio día. En consecuencia, la energía producida por esta fuente es también variable. Interesa, de cara a su uso, tener algún procedimiento para intentar conocer cuánta energía se espera tener para un momento dado a corto y medio plazo, pudiendo servir para identificar si es suficiente para cubrir las necesidades o bien hubiera que hacer uso de otra vía de obtención de energía como puede ser operar la planta de concentración si la hubiera o mirar otra fuente de forma a mantener un suministro suficiente para los consumidores.

Se pretende conseguir una predicción de la energía que se puede producir a corto plazo en una granja solar en base a la radiación solar recibida. La predicción se va a apoyar sobre datos del pasado. En función del enfoque se pueden obtener de distintas maneras como por ejemplo mediante captura y tratamiento de imágenes satélite o con una cámara de cielo en la zona de interés, por mencionar unos ejemplos. Un sistema Internet de las Cosas (Internet of Things, IoT) ofrece otra posibilidad de obtención de la información necesaria mediante una red de sensores. Para ello se despliegan varios

nodos cubriendo la región de las placas solares con el objetivo de recopilar información sobre la radiación recibida en la zona, así como otros parámetros como la temperatura con una cierta frecuencia; mencionar que el despliegue no es el ámbito de este trabajo. Estos datos serán enviados a una unidad de cómputo de mayor potencia para el análisis y la predicción mediante modelos estadísticos, presentando un entorno de Internet de las Cosas.

1.2. Objetivos

Existen trabajos previos (Rincón, 2018) que hacen uso del aprendizaje máquina para generar modelos con los que predecir la energía que se va a generar. El aprendizaje pasa por un entrenamiento que hace uso de los datos que se consideran relevantes.

A partir del mismo conjunto de datos iniciales se pueden obtener diversos modelos en función de los criterios para elegir los datos. Entre ellos se puede destacar las estaciones de medición que se tienen en cuenta, el número de muestras por cada estación, intervalos de tiempo como por ejemplo todo el año, las distintas estaciones, los meses, etcétera. Esto da lugar a modelos con un mayor o menor grado de especificidad, pudiendo presentar en los más genéricos un ruido por parte de los datos que sean menos relevantes así como incrementar el cómputo en tiempo y requisitos de memoria si ha de procesar más datos de lo estrictamente necesario.

El objetivo de este trabajo es explorar el impacto en la precisión de la predicción que tiene hacer uso de una batería de modelos cubriendo una serie de escenarios en lugar de uno único. La idea consiste en conseguir incrementar la precisión buscando relaciones entre los datos de los nodos que indiquen cuantos datos y de qué nodos es relevante seleccionar para predecir la radiación en una estación previamente fijada en función de ciertas franjas temporales.

1.3. Plan de trabajo

Por un lado se realiza una exploración en base a obtener las correlaciones cruzadas entre los datos de todas las estaciones con el objetivo de conocer la afinidad entre ellas y seleccionar los datos en base a este criterio. La radiación solar captada por las placas puede verse afectada por las condiciones climatológicas del momento como vientos que desplazan cúmulos de nubes. En consecuencia, adicionalmente a usar el conjunto global de datos, se estudia la existencia de correlaciones en base al tiempo como pueden ser las estaciones del año (invierno, primavera, verano, otoño), los meses, o franjas horarias concretas como las mañanas.

Adicionalmente y como complemento a la exploración anterior donde se

establece de forma casi manual los distintos modelos, se recurre a la técnica de PCA y posterior clasificación en categorías no predefinidas resultantes de todos los datos para definir los distintos modelos a crear. Este aprendizaje máquina no supervisado permite una valoración de la clasificación elegida.

Capítulo 2

Predicción solar

Las placas solares permiten transformar la irradiación solar en energía eléctrica. Su cantidad e intensidad están sujetas a las condiciones climáticas ambientales de la zona. Así que se ve afectada por los vientos, la calidad del aire, la densidad de los cúmulos nubosos o la inclinación y distancia de la Tierra respecto al Sol por ejemplo. Este aspecto confiere a la energía solar una variabilidad en su obtención. En consecuencia es necesario establecer algunos indicadores para tener una idea de la cantidad que se puede producir en el corto y medio plazo: es decir se busca la predicción de irradiación solar.

La predicción solar se puede estimar a día de hoy con distintas técnicas, y con distinto grado de precisión. Es posible agruparlas en dos categorías: simulaciones numéricas y modelos por un lado frente a métodos de seguimiento de las nubes. De ésta categoría se puede mencionar la captura de imágenes desde satélites o bien fotografiar el cielo desde la superficie terrestre para extraer las características de las nubes en cuanto a tipo así como la dirección y velocidad con la que se desplazan. Respecto a ejemplos de la primera clase nombrada están los modelos matemático-físicos o los modelos estadísticos.

Una opción para los modelos estadísticos es su creación a partir de la recopilación de datos de irradiación mediante una red de nodos por la zona de interés, una alternativa IoT que es la empleada en este trabajo.

2.1. Datos para el modelo

2.1.1. Red de sensores

Un modelo estadístico requiere para su construcción tener un conjunto de datos. En el caso de la predicción solar, es preciso recoger los datos de irradiación de la zona de interés. Para ello se despliega una red de estaciones de medida (nodos) con sensores, en particular de temperatura y humedad que capturan los parámetros y los envían a un servidor central donde se

almacenan antes de ser usados por éste u otro elemento.

Existe variedad de modelos de nodos piranómetros con un coste elevado con lo que no es fácil tener una red muy densa. Por este motivo, se está desarrollando un nodo cuyo coste sea bastante inferior sin que por ello se tenga una pérdida significativa en la calidad de las mediciones. Con ello se puede tener más puntos de recogida de datos con los que trabajar, pudiendo mejorar la precisión de la predicción.

El diseño original de Cebreiros (2017) tiene nodos que cuentan para su despliegue con una batería recargable y una placa solar como fuente de alimentación y autonomía. En cuanto a sensores se han elegido uno de para la radiación (el principal objetivo del nodo), otro para monitorizar la temperatura y humedad internas (permitiendo vigilar las condiciones del nodo pues se espera que esté expuesto al sol cuyas temperaturas en horas centrales particularmente puedan ser lejos de óptimas para el funcionamiento de los componentes), un sensor de temperatura externa y GPS que indica la posición para situar donde está cada nodo, así como los datos que recoge y eventualmente controlar que su posición sea la esperada.

El procesamiento se realiza mediante una LoPy, que permite un mínimo cómputo así como almacenamiento y transmisión de los datos en varias modalidades, teniendo preferencia aquella de bajo coste computacional y energético, con poca carga de datos como nodo de Internet de las Cosas. Cebreiros (2017) ha diseñado una placa que permite la conexión de todos los sensores y batería con la LoPy.

Fisicamente, como muestra la figura 2.1 en este punto se tiene un pequeño contenedor preparado para exterior para la LoPy y los sensores internos. Sobre esta caja se ha diseñado e impreso en 3D un soporte con un grado de libertad para ajustar la inclinación de la placa solar de carga de la batería al valor correspondiente a la ubicación donde se despliega; durante pruebas por el campus universitario se tendría que ajustar a cuarenta y dos grados. El soporte impreso cuenta además con una base para ubicar el dispositivo GPS y la placa que va a tomar las medidas de radiación solar.

2.1.2. Red de Oahu

Debido a que los nodos están en fase desarrollo, no se dispone de una colección de datos suficiente, que abarque más de un año de mediciones. Por ello se ha decidido usar unos datos en abierto (open data) proporcionados por el Laboratorio Nacional de Energías renovables (National Renewable Energy Laboratory, NREL), un organismo que forma parte del Departamento de Energía de Estados Unidos (U.S. Department of Energy).

Estos se corresponden a las mediciones de irradiación en las cercanías del aeropuerto de la isla de Oahu perteneciente al archipiélago de Hawai como se puede ver en la figura 2.2. El conjunto de datos contiene mediciones

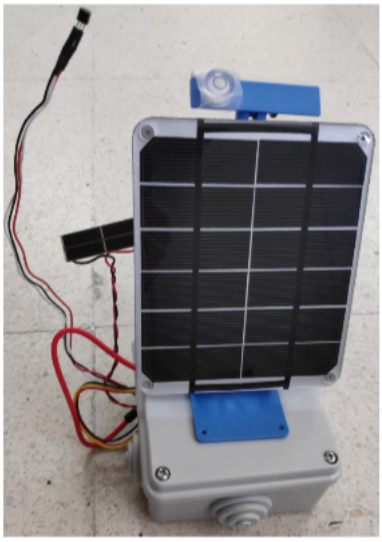

Figura 2.1: Imagen del nodo montado con su caja y soporte, foto de Palomino y Pérez (2019)

provenientes de diecisiete estaciones de medida (nodos) que han registrado en intervalos de cada segundo la irradiación recibida durante el periodo de unos diecinueve meses comprendidos entre el dieciocho de marzo de dos mil diez y el treinta y uno de octubre de dos mil once. Esto permite trabajar con series temporales de irradiación en diecisiete puntos de la región de estudio.

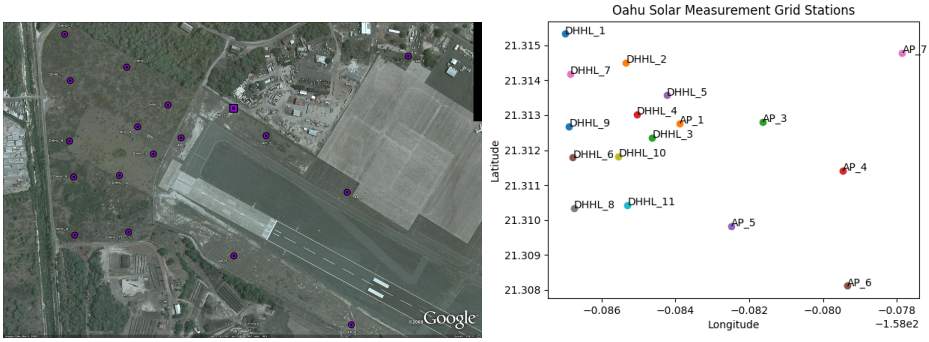

Figura 2.2: Imagen satélite de la región (izquierda) y mapeado de las estaciones (izquierda)

Estos datos han sido preprocesados con el tratamiento que expone Rincón (2018). Cabe destacar una limpieza para tener en cuenta medidas válidas como la limitación al rango horario diurno del día más corto del año (con el objetivo de poder trabajar con todos los días en igualdad de condiciones)

así como la normalización de las mediciones de irradiación con respecto al modelo de cielo claro (Clear Sky Model, CSM).

De las estaciones, una ha tenido que ser excluida para la predicción. Ésta se corresponde a la más cercana a la pista de despegue y aterrizaje que, estando así expuesta a mayores cambios ambientales en sus inmediaciones presenta una cantidad de datos erráticos suficientemente grande para perturbar el modelo resultante pues es el único punto que presenta tales variaciones y la escala espacial no da lugar a microclimas que justifiquen el uso de este nodo.

2.1.3. Características (features)

Los datos que se proporcionan para el modelo (tanto creación como posterior uso) son un conjunto de variables cuyos valores constituyen cada una de las entradas con las que el modelo va a ser entrenado, así como la entrada con la cual se va a realizar la predicción. Estas variables constituyen las características (o features) del modelo. Cada uno de los nodos proporciona las mediciones de radiación solar de la zona, que conforman el conjunto de datos. En el periodo de muestreo disponible se tiene la evolución en función del tiempo de la irradiación en cada estación. Cada uno de estos históricos serán las variables y por consiguiente las características de todo modelo creado a partir de ellos.

2.2. Modelos

2.2.1. Modelo predictivo

La estructura elegida que se ha entrenado con los datos para la obtención del modelo predictivo es una red neuronal. Esta está definida por tres capas de trescientas neuronas (unidad básica de la red) cada una más una adicional para el único valor de irradiación solar que se predice, en la estación de medida previamente seleccionado. Cada una de las capas, excepto la última, hace uso de la función de regresión lineal para ir ajustando los coeficientes de la red en el entrenamiento.

La estructura más básica de red sería tener exclusivamente una neurona por feature, en este caso la red contaría con dieciséis neuronas. Esto implica que para la predicción solar en un punto sólo se tiene en cuenta el conjunto de mediciones en un instante de tiempo seleccionado (ya sea un segundo antes, treinta segundos, un minuto, etcétera). Sin embargo, existen más posibilidades que pueden mejorar la precisión de la predicción como elegir por cada nodos varias mediciones de irradiación, siendo ese número igual para todos los nodos o distinto.

2.2.2. Modelos especializados

El trabajo de Rincón (2018) ha usado para su modelo la totalidad de los datos, recopilados durante un periodo de más de un año. Esta ventana de tiempo es suficientemente grande como para cubrir distintas condiciones ambientales. Es sensato considerar la posibilidad de la presencia de patrones o evolución en los niveles de irradiación característicos de cada caso. La posición de la tierra respecto al sol es distinta según la estación (invierno, primavera, verano, otoño) en la que se esté con lo que los niveles de radiación podrían verse modificados acorde. En función de la localización en la superficie terrestre, se pueden dar fenómenos recurrentes según el momento del día como por ejemplo nieblas al amanecer o calimas vespertinas, que en caso de darse van a disminuir la irradiación recibida.

En consecuencia, se quiere explorar con este trabajo el impacto en la predicción solar que tiene considerar varios modelos con un conjunto de datos más reducido pero dedicado a un espacio temporal concreto o algún fenómeno ambiental característico de la zona de interés comparado con un único modelo global a partir de la totalidad de los datos.

Capítulo 3

Entorno de Desarrollo

Para la predicción de la radiación solar se ha optado por hacer uso de los datos que se recopilan en cada punto de medida mediante algún tipo de aprendizaje máquina supervisado, en particular las redes neuronales, con lo que el entorno de desarrollo tendrá que dar un buen soporte a esto, a ser posible eficiente y de fácil uso.

Siendo el objetivo la aplicación y no su desarrollo, se han comparado algunos entornos de desarrollo que soportan las redes neuronales para encontrar el más adaptado al trabajo.

De todos los existentes se han considerado en la comparativa Scikit-learn, CAFFE y Tensorflow. La toma de decisión se basa en los criterios de requisitos del sistema, los tipos de modelos de aprendizaje máquina que permite desarrollar así como el entorno de programación y la facilidad de uso.

Scikit-learn ha sido empleado en trabajos anteriores (Rincón, 2018), con lo que se ha considerado como referencia así como guía a la hora de filtrar los entornos de desarrollo posibles. En consecuencia los tres posibles candidatos se basan en Python como lenguaje de programación.

Las características potencialmente relevantes y de relativamente fácil acceso se han recopilado en las tablas 3.2, 3.1 y 3.3 en base a la información disponible en la documentación oficial de los entornos. En Particular de CAFFE (Jia, b) y (Jia, a); para Tensorflow (Tensorflow, a) o (Tensorflow, b) y para Scikit-learn (scikit-learn developers).

A nivel del entorno, tanto TensorFlow como Scikit-learn sólo necesitan en el código Python incluir sus librerías. Lo que, una transición entre ellos (estando el código de Rincón (2018) en Scikit-learn) no debería suponer grandes esfuerzos. Sin embargo en CAFFE, sin una toma de contacto, desde la documentación se presentan varias modalidades de uso, todas ellas requiriendo que adicionalmente se tenga en formato Protobuf los datos, la definición de la red (sus capas, método de resolución, etcétera).

Una vez con ello se puede hacer uso de la línea de comandos para crear el modelo; o bien desde Python se pueden cargar los modelos así como que

se encargue de la entrada salida o permitir visualizar las redes neuronales; por último una fuerte integración con Matlab con varias opciones de uso y analisis como se figuran en la tabla 3.2.

Considerando exclusivamente la vía de Python el salto con Scikit-learn no aparenta ser muy grande pero es superior al que hay con TensorFlow de entrada principalmente por pasar buena parte de la definición del modelo (red, datos, etcétera) de Python a Protobuf. En consecuencia, aunque se ha tenido como candidato, no era el primero para hacer las pruebas.

Adicionalmente, como se ha recogido en la tabla 3.3, algunos tipos de redes potencialmente interesantes de CAFFE están presentes en Tensorflow. Es relevante destacar que sólo en la documentación de TensorFlow y Scikit-learn está bien visible el soporte para modelos de regresión, que es el que se ha utilizado en Rincón (2018).

A nivel de requisitos (tabla 3.1), son bastante similares, orientados a Unix, pero también en otros sistemas operativos. De los grandes, destacar el caso de Windows, en el que Tensorflow es el mejor adaptado al instalarlo casi directamente como figura en su documentación. Sin embargo los otros dos toman un enfoque ligeramente más externo. Scikit-learn necesita del plugin WinPython, un proyecto aparte mientras que CAFFE es una ramificación del poryecto original. Ésto crea dudas respecto a garantizar trabajar con la versión del entorno adecuada y sobre la aplicación de eventuales actualizaciones en el proyecto principal.

En base a la cercanía, con el primero como se presenta en la tabla 3.2 para el entorno y en la tabla 3.1 respecto a los requisitos del sistema donde se ejecuta, se hicieron pruebas en Tensorflow replicando el trabajo anterior de Rincón (2018) con el objetivo de poder comparar: una red neuronal de tres capas con tres cientas neuronas por capa.

En el proceso de familiarización con Tensorflow para su evaluación siguiendo guías como Bourguignat al igual que la documentación del mismo (Maxim, Tensorflow (b)), se encontró el entorno Keras, una capa por encima de Tensorflow facilitando su uso (aunque también admite otras herramientas equivalentes). Debido además que los únicos requisitos son los de Tensorflow (keras team, a) en la modalidad que se ha elegido, no se ha incluido en las tablas, pues es una forma de hacer uso de Tensorflow.

Para que fuera lo más fielmente posible, se miró en detalle la información de la documentación de scikit-learn (scikit learn) y de keras (keras team (b), scikit learn) para establecer todos los valores de Rincón (2018) en Keras.

En base a una curva de aprendizaje facil con Tensorflow para los primeros pasos así como una transición rápida desde Scikit-learn, y con la abstracción añadida por Keras al simplificar lo que se tiene que realizar en Scikit-learn, Tensorflow ha destacado con diferencia entre los tres entornos que se han estado considerando. Una vez replicado el modelo de Rincón (2018), al tener unos resultados casi idénticos en las métricas y la predicción, así como una

Scikit-learn	Unix OS, Ubuntu 16 o 14, Windows (WinPython), Python(>=2.7 o >=3.3), NumPy(>=1.8.2)
TensorFlow	Ubuntu 16.04–12.04, Si GPU: NVIDIA, WIndows 7 min
CAFFE	Ubuntu 16.04–12.04, Si GPU: CUDA, BLAS o OpenBlas, Boost(>=1.55) Opcional: OpenCV (>=2.4), librerías E/S:lmdb, leveldb(snappy), cuDNN (GPU) Python Caffe: Python(>=2.7 o >=3.3), Numpy (>=1.7) boost-provided: boost.python, make o CMake

Tabla 3.1: Comparativa de los entornos de desarrollo: requisitos del sistema donde se va a ejecutar

gran variedad de otros modelos de aprendizaje máquina, de posible interés para trabajos posteriores, se ha concluido realizar el trabajo bajo Keras con soporte Tensorflow.

Scikit-learn	Código Python (librerías scikit-learn)
TensorFlow	Código Python (librerías Tensorflow)
CAFFE	Datos en Protobuf: definición de red, capa, método de resolución Línea de comandos: modelo, entrenamiento, evaluación,estado Python: cargar modelos, propagación adelante-atrás, E/S, visualizar redes Matlab: múltiples redes, acceso y modificación en todo punto de la red con obtención/edición de datos cualquier método resolutivo, retomar resolución desde captura ejecutar nº fijo de iteraciones, Mezcla Matlab y etapas en gradiente

Tabla 3.2: Comparativa de los entornos de desarrollo: entorno

Scikit-learn	SVN: no lineal,lineal, regresión(vector), Stochastic gradient descent, Nearest neighbours regression, Gaussian regression (classification), Decision trees regression, Gradient tree boosting regression, Neural networks model regression, Neural networks unsupervised
TensorFlow	Red profunda, Deep residual network, Baseline, Deep neural network regressor, Linear regression, Convolutional neural network, Logistic regressor
CAFFE	Red profunda, Red recurrente, Red convolucional contenida, Region proposal network, Red convolucional

Tabla 3.3: Comparativa de los entornos de desarrollo: modelos

Capítulo 4

Análisis de correlación

El conjunto de datos cubre mediciones a lo largo de todo el día y al menos un ejemplo de cada día del año. Previo filtrado, para tener en cuenta las horas diurnas presentes en todas las fechas entre otros, trabajos anteriores (Rincón, 2018) han utilizado la totalidad de los días disponibles para sus modelos.

Pudiendo existir fenómenos o tendencias temporales a lo largo del año, por el cambio de estaciones o incluso a lo largo de los días como pudieran ser nieblas al amanecer reduciendo la irradiación solar recibida comparado con el resto de la jornada, se ha propuesto explorar distintos escenarios para obtener redes neuronales expuestos a datos correspondiendo exclusivamente a ciertos periodos de tiempo más reducido. Con esto se ha buscado la posibilidad de construir modelos más específicos con vistas a lograr mejores resultados de predicción. Adicionalmente, por hacer uso de un subconjunto de los datos, se debería reducir el coste de creación del modelo en tiempo de ejecución y memoria.

Debido a la cantidad de datos disponibles, así como la gran cantidad de posibles casos en base a los fenómenos meteorológicos, se ha decidido acotar la exploración a los elementos temporales de las estaciones anuales (invierno, primavera, verano y otoño), los meses, y una separación en tres franjas horarias equilibradas respecto al número de muestras: mañana, mediodía y tarde.

Con esta criba se tiene una cobertura de casuística típica pero sin ser por ello exhaustiva. En base a los datos disponibles se ha aplicado un segundo criterio para la creación de los modelos con mayores probabilidades de obtener un mejor resultado que en los trabajos anteriores. Esta selección reposa en la aplicación de correlación cruzada sobre los datos.

La correlación cruzada permite obtener la similitud entre dos variables. Un valor alto implica una fuerte relación entre ambas. Este valor puede ser positivo o negativo, lo que permite conocer si existe una fuerte relación con un desfase, pudiendo ser temporal si es una de las dimensiones a tener en cuenta en el sistema con el que se trabaja como es el caso de este estudio sobre la radiación solar; así como la diferencia de tiempo que puede existir entre

los valores captados por las distintas estaciones (Firing, 2019). En base a la implementación usada, la primera variable será considerada la de referencia, y la segunda con la que se compara. Si el resultado de la correlación cruzada es un valor positivo significa que la segunda señal presenta los valores que en un tiempo posterior van a estar presentes en la de referencia; se da la relación opuesta entre las señales para una correlación negativa.

Aplicando esta técnica a los datos de entrada de las mediciones de la radiación solar se busca seleccionar un subconjunto de muestras de las estaciones (pudiendo eliminar estaciones por completo incluso). Las estaciones (con sus varias medidas en el pasado) son las características (features) que se usan en el modelo de red neuronal para la predicción de la radiación solar. El número de características influye en el modelo, tanto creación como aplicación, dirigiendo los resultados al igual que repercutiendo sobre el tiempo de ejecución.

Eligiendo un número de muestras fijo en el pasado para cada una de las estaciones implica que haya una cantidad elevada de características y no todas aportarían necesariamente información útil. llegando incluso a generar ruido, sin contar con la sobrecarga computacional, tanto en recursos de memoria como en tiempo, que supone. En consecuencia, se busca encontrar un subconjunto adecuado de características que permitan la predicción de la radiación con una calidad como mínimo similar a la de los modelos actuales, reduciendo así los requisitos en espacio y el tiempo de ejecución. Lo que permitiría el poder usarlo en un entorno IoT donde prima un uso eficiente de los recursos como la reducida memoria y el bajo consumo, expresándose en una baja carga computacional, en comparación con un servidor o un ordenador.

Los datos sobre los que aplicar la correlación cruzada son mediciones cada segundo de cada una de las estaciones en un periodo de casi 20 meses completos consecutivos, habiendo acotado las horas de mediciones entre las 07:30 y las 17:30 por presentar valores válidos en todos los puntos de captura de datos a lo largo de todo el periodo; incluyendo los meses de invierno que han reducido ligeramente el intervalo a esa franja temporal.

Se ha decidido considerar tramos de 15 minutos (900 segundos) para la correlación de forma a tener un número de muestras mínimo con las que poder trabajar sin por ello hacer uso de la totalidad que llevaría las correlaciones más cercanas a 0 por la variabilidad de los valores. En un primer tiempo se hizo uso de los datos para modificarlos los menos posible; sin embargo, debido a algunos periodos de alta variabilidad, pudiendo deberse a ruido se ha realizado la correlación en el periodo de 15 minutos, pero cada dato se corresponde a la media aritmética de diez muestras de un segundo originales. Al aplicar la correlación cruzada se obtiene el valor de la correlación entre ambas señales en el intervalo [-tramo; tramo], en este caso [-15;+15] minutos, correspondiendo a la afinidad entre las señales para cada uno de los desfases temporales.

Los resultados de la correlación se van acumulando en base a varios criterios potencialmente relevantes, para explorar la viabilidad de los modelos pensados, creando histogramas de las correlaciones en función del desfase temporal. Estos histogramas son generados para cada par de estaciones por separado; en los de dh3 y dh5 solo van a estar las correlaciones de estas dos estaciones. Se ha considerado que pueda ser interesante un histograma para todos los datos de los casi veinte meses, a título comparativo para poder evaluar mejor los otros resultados.

Adicionalmente se ha planteado la separación en base a las estaciones del año (invierno, primavera, verano, otoño) y por cada mes. A su vez, por cada mes se ha elegido contabilizar por un lado el mes completo y por otro lado separarlo en tres franjas horas de mañana, mediodía y tarde en el supuesto que hubieran fenómenos meteorológicos que se presenten bajo alguno de los escenarios contemplados y tengan repercusión sobre la relación de las medidas de radiación entre las estaciones.

Una representación gráfica de los histogramas permite observar unas correlaciones positivas en las estaciones del Este respecto a las del Oeste, en buena parte de los casos contemplados, comportamiento en concordancia con los vientos, predominantes en la zona con dirección de Noreste a Suroeste (boreal naciente a austral poniente).

A pesar de haber diferenciado unos pocos escenarios, se han obtenido (4*12+4+1)*17 = 901 histogramas a razón de los cuatro por cada mes (total del mes, mañanas, mediodías y tardes) más las estaciones y el total por cada una de las estaciones, sólo haciendo uso de una única estación de referencia, en este caso la ubicada relativamente en el centro dh3. En consecuencia, y con la finalidad de obtener parámetros más objetivos para la evaluación de los histogramas de correlación, se ha optado por recopilar unas métricas descriptivas de éstos en lugar de la apreciación visual de la representación gráfica.

Las métricas elegidas se reducen a cinco. El valor medio ponderado de los desfases temporales (x), la desviación estándar respecto a esta media (y), el área cubierto por la desviación estándar (z) para obtener la parte de datos cubiertos incluidos en esa zona, así como la media (t) y el máximo de las correlaciones para el área de la desviación estándar (u).

Dada una estación objetivo (A), la media, cuanto mayor sea su valor, no solo absoluto, sino de signo positivo también, mejor será la estación de comparación (B) para predecir los valores de A. La desviación estándar, interesa que sea un valor reducido pues refleja que los desfases temporales con mayor correlación están próximos al desfase medio. A mayor área z, más peso tienen los desfases incluidos por la desviación estándar. Una media t y máxima u elevadas muestran un alto peso de los desfases próximos al desplazamiento temporal medio.

El haber hecho uso de la media aritmética de diez valores en lugar de los

datos originales para la correlación, aquellas métricas que hacen referencia al desfase temporal deben multiplicarse (como sería el caso de la variable y) por diez para obtener el valor en segundos real.

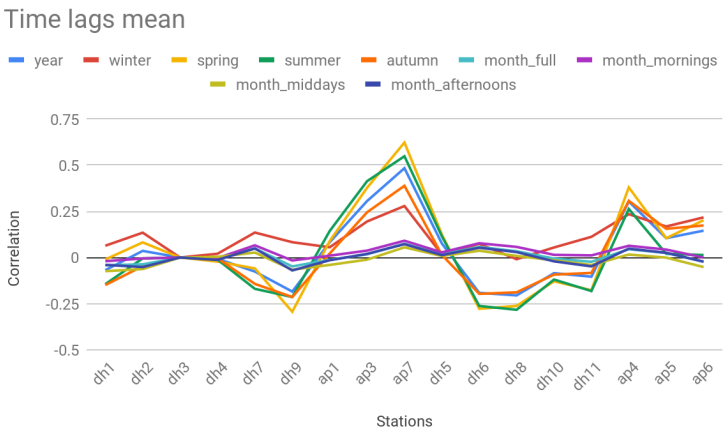

Figura 4.1: Media de diferencias de tiempo respecto a la estación de medida dh3

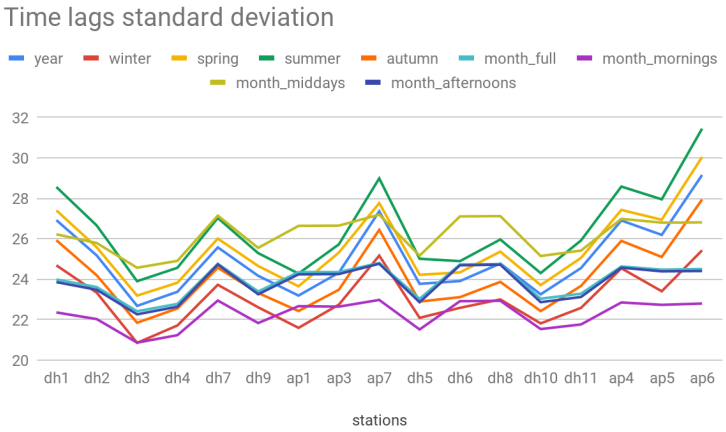

Figura 4.2: Diferencia de tiempo correspondiendo a la desviación estándar (parte de valores positivos)

Entre los escenarios contemplados, se han elegido los dos que pudieran presentar mejoras de forma más distintiva respecto al trabajo anterior (Rincón, 2018). Como se aprecia en las figuras de cada una de las métricas (valor medio de desfases temporales (x) 4.1, desviación estándar respecto a la media (y) 4.2, área de la desviación estándar (z) 4.3, media de las correlaciones para el área de la desviación estándar (t) 4.4, máximo de las correlaciones

winter	mean	standard deviation
dh3_dh1_winter	0.06367314115	24.68570383
dh3_dh2_winter	0.1344376721	23.3340779
dh3_dh3_winter	-7.30E-16	20.8536778
dh3_dh4_winter	0.02011209438	21.71474402
dh3_dh7_winter	0.1351815362	23.72458105
dh3_dh9_winter	0.08279950635	22.6117664
dh3_ap1_winter	0.05531214434	21.59664777
dh3_ap3_winter	0.1950068899	22.74375382
dh3_ap7_winter	0.2787335924	25.16554917
dh3_dh5_winter	0.0166090504	22.09020407
dh3_dh6_winter	0.07403581129	22.58546724
dh3_dh8_winter	-0.009366145596	22.99986853
dh3_dh10_winter	0.05417978485	21.81474136
dh3_dh11_winter	0.111954353	22.58326115
dh3_ap4_winter	0.234625293	24.5395584
dh3_ap5_winter	0.1679754728	23.40385965
dh3_ap6_winter	0.2172400078	25.42763508

Tabla 4.1: Modelo de invierno: valor medio y desviación estándar

months_mornings	mean	standard deviation
dh3_dh1_morningmean	-0.02008233291	22.35896772
dh3_dh2_morningmean	-0.005365813403	22.02829043
dh3_dh3_morningmean	-3.83E-17	20.86535207
dh3_dh4_morningmean	-0.00319764247	21.23492766
dh3_dh7_morningmean	0.06586197349	22.94447356
dh3_dh9_morningmean	-0.01631204648	21.83410067
dh3_ap1_morningmean	0.009849457998	22.66708989
dh3_ap3_morningmean	0.03724222387	22.64476137
dh3_ap7_morningmean	0.09076674762	22.97607666
dh3_dh5_morningmean	0.02781673439	21.5160911
dh3_dh6_morningmean	0.07716873776	22.91434644
dh3_dh8_morningmean	0.05731940777	22.9347496
dh3_dh10_morningmean	0.01443071081	21.53580048
dh3_dh11_morningmean	0.01137096064	21.76458576
dh3_ap4_morningmean	0.0635025374	22.85152436
dh3_ap5_morningmean	0.04290665407	22.73376653
dh3_ap6_morningmean	-0.0004759253914	22.79729268

Tabla 4.2: Modelo mañanas: valor medio y desviación estándar

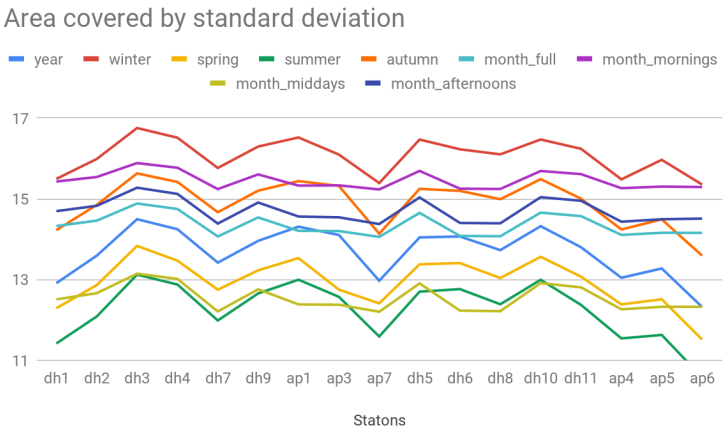

Figura 4.3: Area covered by standard deviation

para el área de la desviación estándar (u) 4.5) los modelos de "invierno" por un lado y, "exclusivamente las mañanas de todos los días" por otro, son los más prometedores.

En las tablas 4.1 y 4.2 se presentan la media y la desviación estándar de estos dos escenarios. Los valores de la variable (y) se corresponden con el número de muestras elegidos por cada estación a la hora de construir la matriz de datos de entrada necesaria en la creación de los modelos debido a que cubren la mayor parte de la correlación positiva. Esto se debe a que las muestras de datos son cada diez segundos, al igual que el tratamiento de correlación, con lo que los resultados de este proceso dan el número de medidas a elegir (usando el valor entero más próximo) como aparece en la tabla 4.3. En ella figuran la cantidad de medidas de irradiación por cada nodo en función del modelo. Estando ordenadas alfabéticamente las estaciones de medida se percibe por un lado la ausencia de ap2, debido a que es la estación problemática comentada anteriormente en el apartado 2.1.2.

Otro elemento a destacar es respecto a ap3. La falta de muestras se debe a que los datos que ha registrado presentan un comportamiento similar a ap2. Al no ser tan pronunciado no se ha eliminado en fases anteriores. Todas las estaciones cuentan con un número de días útiles comprendido entre 593 y 595, sin embargo ap3 presenta 409. Una vez elegido el número de muestras, se crea la matriz con los datos que se van a usar para el modelo teniendo en cuenta entre otros factores que el número de días sea el mismo para cada nodo evitando tener que inventar valores o hacer inferencias. En consecuencia, hacer uso de ap3 elimina en torno a un tercio de los datos, cuyo efecto no es en absoluto despreciable, ni en el entrenamiento ni en la predicción de los modelos.

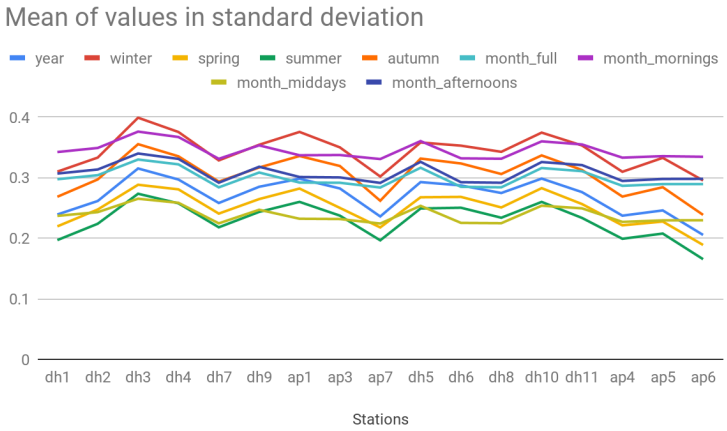

Figura 4.4: Average standard deviation

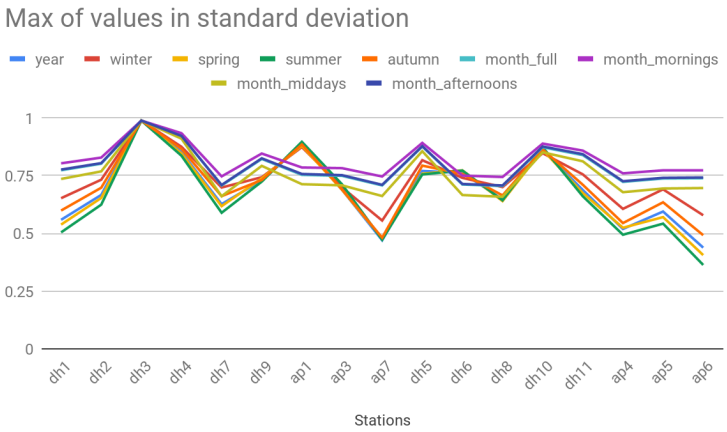

Figura 4.5: Maximum values of standard deviation

Estación	mañanas	invierno
ap1	22	21
ap3	0	0
ap4	22	24
ap5	22	23
ap6	22	25
ap7	23	23
dh1	22	24
dh2	22	23
dh3	20	20
dh4	21	21
dh5	21	22
dh6	22	22
dh7	22	23
dh8	22	23
dh9	21	22
dh10	21	21
dh11	22	22

Tabla 4.3: Modelos seleccionados con el número de muestras a tener en cuenta por cada estación: mañanas e invierno

Capítulo 5

Clustering para modelado

5.1. Método de reducción de componentes PCA

El método de reducción de componentes (PCA: principal component analysis) tiene como aplicación simplificar el problema que se está tratando, lo que permite en este caso obtener un procesado de los datos en menor tiempo.

En el campo del aprendizaje máquina, se tienen los datos de entrada en el formato de vectores con tantos elementos como características (features) a considerar. Cada feature se comporta como una dimensión a tratar por el algoritmo para configurar el modelo. Éste, en su creación, se tiene que adaptar para tener en cuenta todas las características, extrapolando algún patrón, idealmente sin ajustarse en exceso a los datos para poder tener mejores resultados.

Si hay un número elevado de features, será más complejo el modelo y la generalización del problema a resolver podrá acercarse más de lo deseado a los datos de entrenamiento; sin mencionar el coste computacional en tiempo y espacio.

Una alternativa a reducir manualmente características, pudiendo sin ser consciente quitar alguna importante, consiste en hacer uso de PCA. Al aplicarlo se proyectan las m features en un espacio de n dimensiones donde m es estrictamente mayor que n, obteniendo en el proceso nuevos valores de irradiación (VanderPlas, 2016).

5.2. Aplicación de PCA

Los datos con los que se ha trabajado proceden de un total de diecisiete estaciones. En el modelo del trabajo anterior (Rincón, 2018), al tener tres mediciones de irradiación por estación, se está ante un caso de 51 dimensiones. Esta hiperdimensionalidad está todavía más pronunciada en los modelos

elegidos mediante correlación cruzada (ver capitulo 4).

Se hace uso de PCA para reducir el número de features de los datos de partida (una vez limpiados y normalizados con el modelo de cielo claro), quitando así posibles datos innecesarios que podrían introducir ruido o bien presentar al modelo con información ya presente en otras características.

Con este nuevo conjunto de vectores, se realiza un proceso con la misma idea que la correlación cruzada vista en el capitulo 4. Mediante la clasificación no supervisada de clustering k-means se establecen los distintos modelos a entrenar a partir de la cercanía y por ende pertenencia a las clases de los vectores de datos reducidos por PCA, aplicación similar a la realizada por Eschenbach (2018).

En la implementación se ha conservado el índice de cada vector para que sirva de clave y poder separar los datos de origen en función de las clases. Teniendo así unos subconjuntos de datos obtenidos automáticamente haciendo uso de los datos que aportan mayor información al haber aplicado PCA.

En el caso de la correlación, al igual que con el trabajo de Rincón (2018), los datos están en secuencia temporal, ordenados por días. Sin embargo, PCA y clustering tratan cada vector independientemente, con lo que algunos días están repartidos entre varias clases. Debido a la importancia de tenerlo en cuenta a la hora de elegir qué vectores coger para la predicción (dentro del mismo día, no los últimos segundos de un día para el primer valor de radiación del siguiente) en el caso de la correlación, parte del código se encarga de comprobar esto.

Adicionalmente, también se garantiza este aspecto en el caso de PCA y clustering al trabajar con los índices durante el entrenamiento. En lugar de ofrecer a la red neuronal los vectores exclusivos de una de las clases, se dan todos los datos y sólo los índices de cada modelo específico a crear. Así el valor de salida a seleccionar entre los datos para comparar con el valor resultante de la predicción se corresponde al horizonte de predicción elegido y no supone ningún salto por no tener los días consecutivos.

5.3. Resultados

La implementación en Python de scikit-learn para PCA y la de k-means permite elegir en el primero el número de componentes relevantes que se quieren obtener, el número de dimensiones a las que reducir los datos de entrada. Respecto al segundo el número de clases se define por el usuario.

La exploración se ha acotado a reducir las dimensiones a dos, tres y cuatro características con el objetivo de intentar apreciar el impacto que tiene respecto a los valores de partida. Para cada uno de estos casos, se ha contemplado una división en dos, tres y cuatro clases. En consecuencia se tienen veintisiete modelos especializados.

De obtenerse mejores resultados de predicción en los modelos comparado con el modelo único convendría primar el uso de los primeros en detrimento del último. En el caso de aquellos obtenidos mediante el estudio de correlación, se elige el modelo concreto a aplicar en función del momento que se quiera predecir. Sin embargo, en el caso de PCA y clustering esta decisión va a ser realizada en base a ejecutar k-means sobre los nuevos datos.

Capítulo 6

Resultados: precisión de los modelos desarrollados

6.1. Métricas

La precisión de los modelos se ha medido en base a varias herramientas. Por un lado se ha creado un modelo base conservador o baseline. Éste se genera de la misma manera que su contraparte a nivel de los datos de entrada que se le suministran. Sin embargo en lugar de realizar el entrenamiento de una red neuronal (en principio de mismas características que el modelo que se está poniendo a prueba), se asigna como resultado de predicción el mismo valor que la última medición de irradiación en base a que las condiciones no suelen cambiar drásticamente en un corto espacio de tiempo.

Para comparar en primera instancia el modelo elegido frente a su baseline y luego frente a otros, en particular el creado a partir de la totalidad de los datos se han usado dos métricas, una dependiente de la otra.

Por un lado, el error cuadrático medio (mean squarred error, mse). Éste indicador establece la distancia media que existe entre los valores predichos por el modelo y el valor que hay entre los datos. Finalmente se trabaja con la raíz cuadrada de este indicador (root mean squared error, rmse).

Esto se aplica tanto al modelo como a su baseline, lo que permite construir la otra métrica. Ésta, denominada skill en este trabajo, da la relación entre los rmse de los modelos con respecto a su modelo base conservador mediante la fórmula:

$$skill = 100 * (1 - (rmse_modelo/rmse_baseline)) \tag{6.1}$$

Si los errores son iguales el skill es cero, y se puede concluir que el modelo no aporta ninguna mejora. Un skill negativo viene por un rmse más alto en el modelo, con lo que éste predice peor que el baseline. Así que lo que se busca es un skill positivo.

6.2. Resultados obtenidos

6.2.1. Resultados correlación

6.2.1.1. Modelo de invierno

La figura 6.1 compara los skills de la predicción con los datos de los días de invierno aplicados en el modelo entrenado con esos mismos datos al igual que con el modelo entrenado con todos los días disponibles, y por comparación figura los skills de predicción de referencia o base (Rincón, 2018). Los valores exactos de los skills se pueden consultar en las tablas 6.1 y 6.2.

Por la gran diferencia para todos los horizontes de predicción entre el modelo base (entrenamiento y predicción) con el modelo exclusivamente de invierno, de 10 unidades o más sobre un máximo conseguido de 40, este modelo no mejora la predicción solar.

Comparando la predicción para invierno con el modelo de referencia, el de invierno sigue teniendo peores resultados aunque la diferencia es menor. Caben destacar los horizontes de predicción 5 minutos, 10 minutos y puede que 30 segundos, donde la diferencia se deba al margen de variabilidad que se puede dar entre ejecuciones.

Antes de concluir la ineficacia del modelo para los días de invierno conviene tener en cuenta que el modelo sólo ha contado con noventa días, de los cuales sólo una porción para entrenar, otra para validar y la última para la predicción. Es un número bastante reducido como para obtener un modelo adecuado, sobretodo cuando se compara con el de referencia cuyo conjunto de datos inicial cuenta con más de quinientos días.

6.2.1.2. Modelo de mañanas

La figura 6.2 compara los skills de la predicción con los datos de los días de invierno aplicados en el modelo entrenado con esos mismos datos al igual que con el modelo entrenado con todos los días disponibles, y por comparación figura los skills de predicción de referencia o base (Rincón, 2018). Los valores exactos de los skills se pueden consultar en las tablas 6.1 y 6.2 al igual que en el caso de invierno.

En este caso los tres casos de predicción se presentan con una diferencia ínfima de hasta 2 unidades excluyendo el horizonte de predicción de 30 segundos, donde el modelo con sólo datos de mañanas en todas las etapas tiene un skill de casi 10 unidades inferior a los otros dos.

Esta pequeña diferencia se explica con el margen de variabilidad entre ejecuciones o a una ligera perdida en la calidad de la predicción.

Salvo el caso de 30 segundos que precisaría de un estudio más en profundidad, este modelo se ve un buen candidato como sustituto o complemento al de referencia debido a la cercanía de los resultados y la mejora a nivel

computacional por haber hecho uso de un tercio de los datos.

6.2.2. Resultados PCA

En las figuras (6.3,6.4, 6.5, 6.6, 6.7 y 6.8) se presentan los skills de la predicción de los modelos obtenidos de la clasificación una vez aplicado PCA comparado con la predicción resultante de ejecutar el modelo de referencia (Rincón, 2018) con el subconjunto de datos correspondiente a cada uno de los modelos obtenidos.

En el eje de las abscisas figuran los distintos modelos según una nomenclatura numérica donde la primera cifra hace referencia al número de clases que se han elegido crear, el segundo al número de componentes (dimensiones) con las que se queda PCA y el tercero el número de la clase concreto empezando en 0. Por ejemplo el 240 indica que hay dos clases, éstas obtenidas al reducir a cuatro dimensiones los datos de irradiación y que se corresponde con la clase 0.

Al igual que en el caso de la correlación cruzada, los valores de los skills están recogidos en las tablas 6.3 para los horizontes de predicción de 10 segundos y 30 segundos, 6.4 para los horizontes de predicción de 1 minuto y 2 minutos y 6.5 para los horizontes de predicción de 5 minutos y 10 minutos. En ellas, al igual que gráficamente en las respectivas figuras (6.3,, 6.5, 6.6, 6.7 y 6.8) se aprecian algunos valores negativos de los skill para todos los horizontes de predicción salvo en el caso de 30 segundos.

Durante el desarrollo puede suceder tener estos resultados, particularmente al ejecutar el entrenamiento de varios modelos simultáneamente en la misma máquina, por conflicto de recursos. Normalmente se subsana al repetirlo de forma individual. Sin embargo los siete escenarios 320, 332, 341, 421, 432, 433 y 441 presentan entre uno y tres skills negativos a pesar de repetir las ejecuciones de forma individual. En consecuencia, estos subconjuntos de datos para los horizontes de predicción afectados no son viables para su uso de forma separada como se ha visto en este estudio sin hacer un analisis en mayor profundidad.

A nivel de los skills, las diferencias permiten extraer que ningún modelo supera el de referencia (Rincón, 2018), puesto que cuando es mayor el específico no llega a distanciarse de una unidad como en el 331 a 10 segundos o casi dos unidades con 431 a 10 segundos. Sin embargo se presentan casos en los que los modelos específicos tienen valores casi idénticos o muy cercanos al de referencia. Dónde mejor se aprecia es al separar en dos clases el conjunto de datos y reduciendo a dos y tres componentes para todos los horizontes de predicción. La separación en cuatro clases a partir de la reducción a dos y tres dimensiones presenta una diferencia mayor que en los casos anteriores pero sigue siendo muy baja a excepción de los modelos 420 y 433 para la mayoría de los horizontes de predicción.

Modelo\Predicción	10 segundos	30 segundos	1 minuto
invierno	29.51737123	19.47260587	22.85232083
mañana	39.41310684	21.7399718	31.01083216
test invierno (train ref)	32.4017206	27.78974798	27.64668746
test mañana (train ref)	41.0020068	31.39934912	33.77355598
ref (train & test)	38.31491648	30.18258102	32.42686136

Tabla 6.1: Skills para los modelos de invierno y mañanas al igual que el de referencia de Rincón (2018) para los horizontes de predicción 10 segundos, 30 segundos y 1 minuto

En vistas a estos resultados crear modelos específicos usando PCA y clasificación es relevante en los casos mencionados próximos al modelo de referencia, por hacer uso de menos datos para conseguir resultados similares, un aspecto fundamental en un entorno IoT.

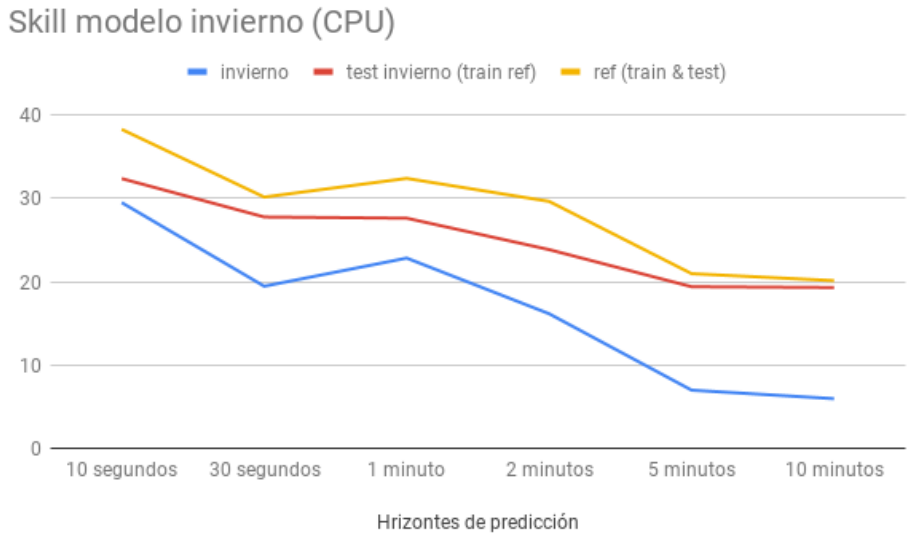

Figura 6.1: Comparativa de los skills del modelo de invierno

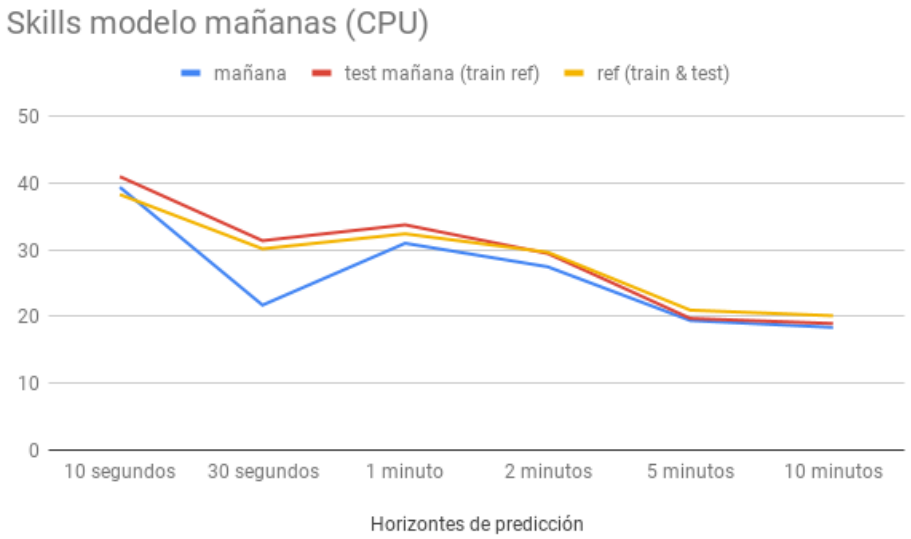

Figura 6.2: Comparativa de los skills del modelo de mañanas

Modelo\Predicción	2 minutos	5 minutos	10 minutos
invierno	16.16720496	6.986708835	5.96444152
mañana	27.48367585	19.42859757	18.39986255
test invierno (train ref)	23.87041961	19.41165485	19.3176039
test mañana (train ref)	29.47864078	19.74092725	18.97805661
ref (train & test)	29.66320887	20.98180842	20.14626761

Tabla 6.2: Skills para los modelos de invierno y mañanas al igual que el de referencia de Rincón (2018) para los horizontes de predicción 2, 5 y 10 minutos

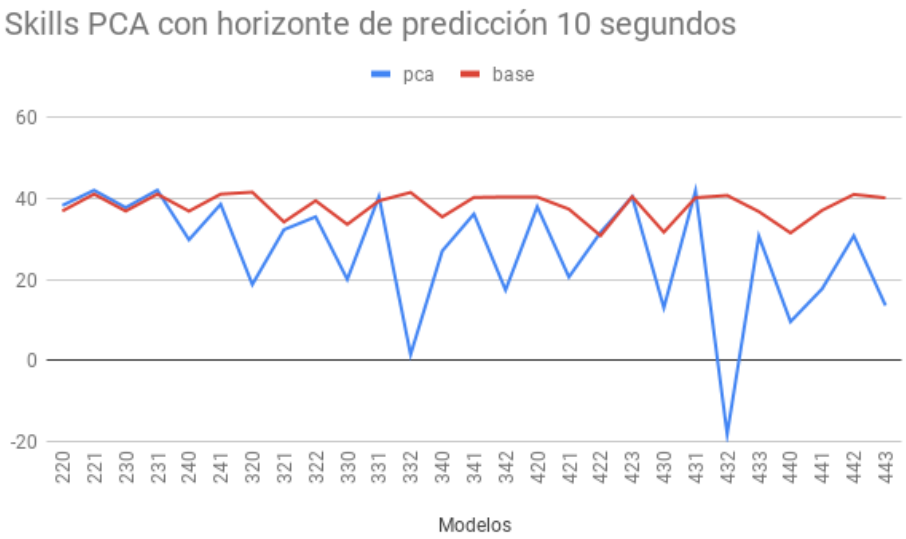

Figura 6.3: Comparativa de los skills para los modelos de PCA con el horizonte de predicción de 10 segundos

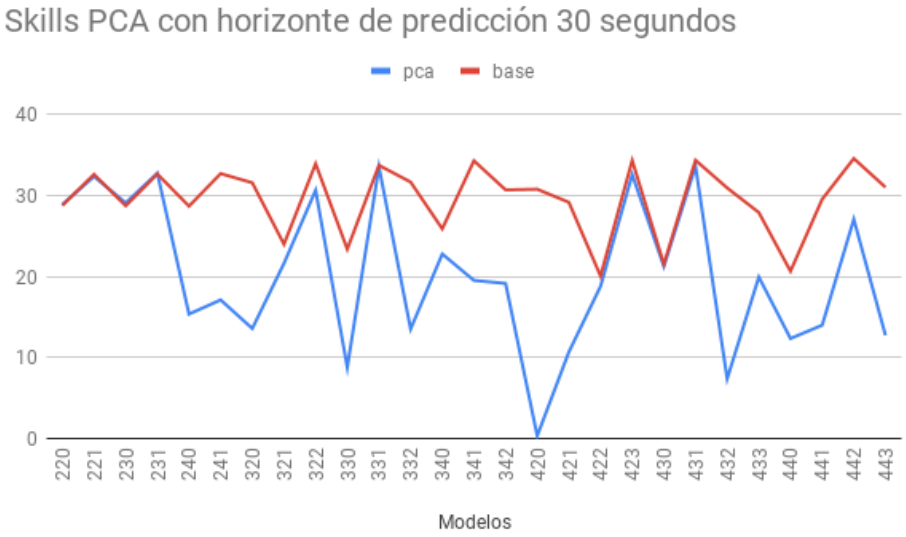

Figura 6.4: Comparativa de los skills para los modelos de PCA con el horizonte de predicción de 30 segundos

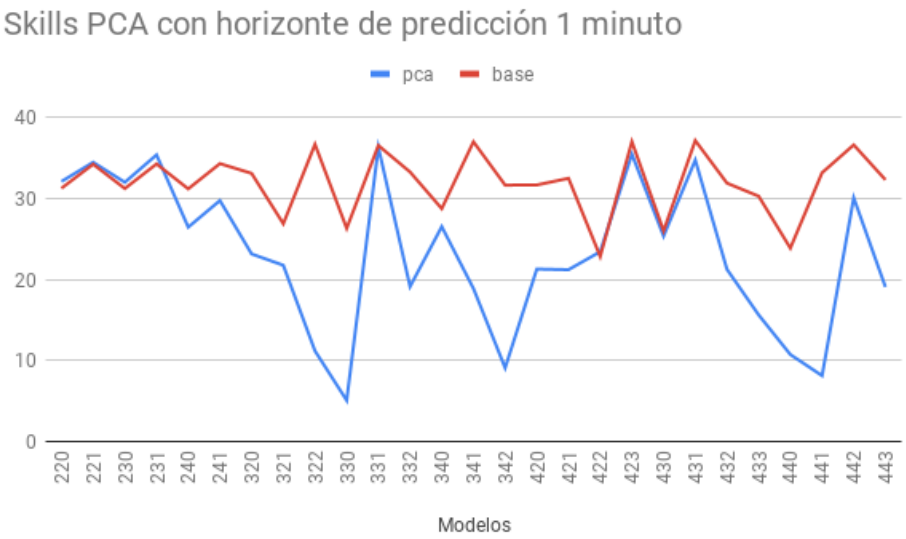

Figura 6.5: Comparativa de los skills para los modelos de PCA con el horizonte de predicción de 1 minuto

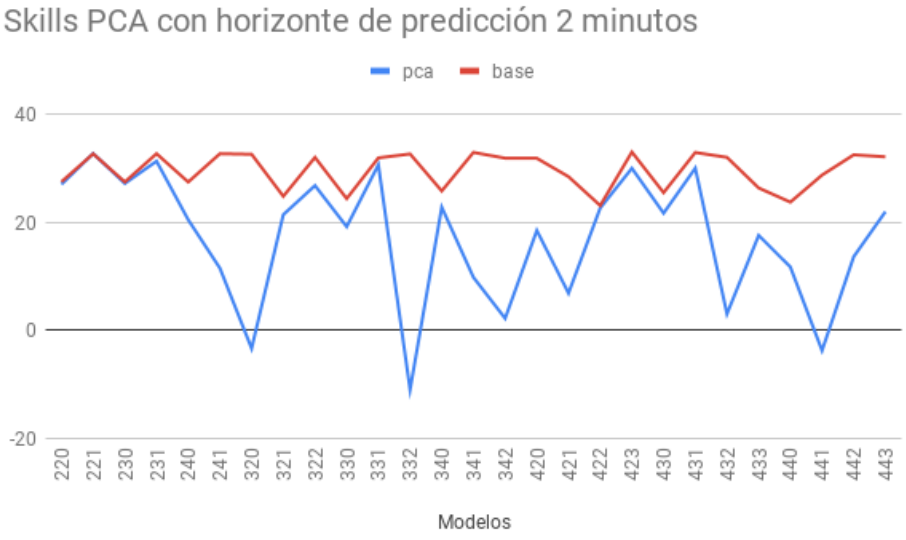

Figura 6.6: Comparativa de los skills para los modelos de PCA con el horizonte de predicción de 2 minutos

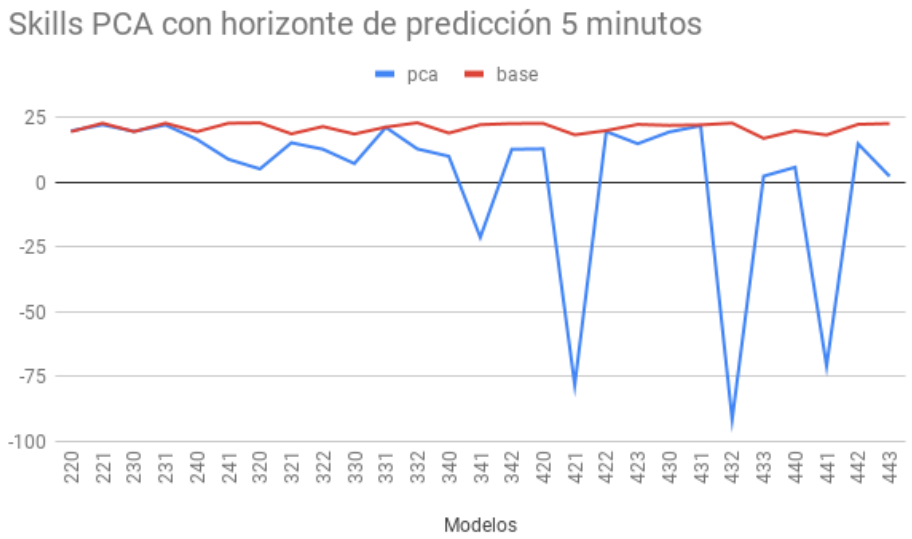

Figura 6.7: Comparativa de los skills para los modelos de PCA con el horizonte de predicción de 5 minutos

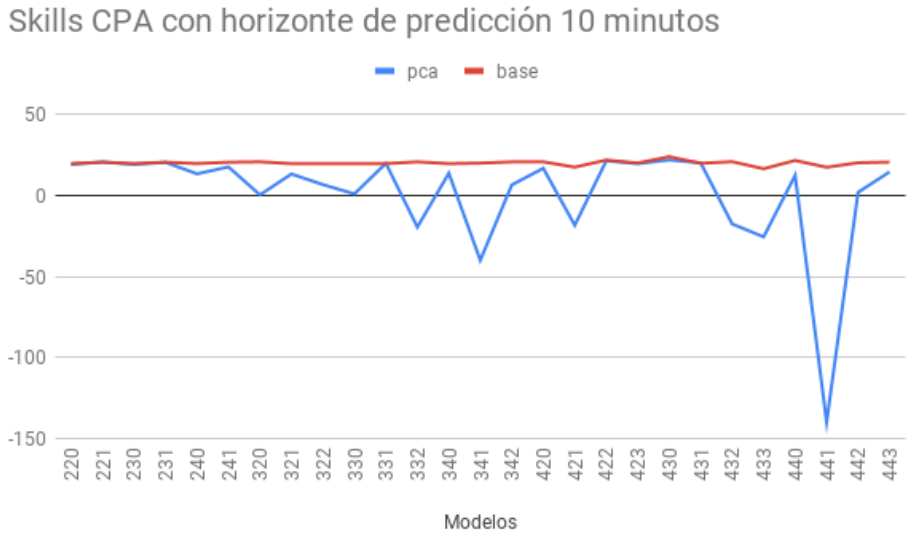

Figura 6.8: Comparativa de los skills para los modelos de PCA con el horizonte de predicción de 10 minutos

Modelo	10 segundos	10 segundos	30 segundos	30 segundos
	pca	base	pca	base
220	38.30440558	36.93314916	28.90704646	28.77588609
221	42.05515049	41.10262719	32.35200286	32.62614841
230	37.77999854	36.9180842	29.09585084	28.73586468
231	42.07665878	41.10201036	32.77123509	32.66883531
240	29.82985626	36.89196318	15.36439265	28.69145206
241	38.5929076	41.102309	17.11796722	32.71334591
320	18.77615318	41.5837104	13.58696343	31.58287205
321	32.32188566	34.23187169	21.62864957	23.99608543
322	35.50285452	39.47244796	30.64701929	33.92733655
330	20.04027995	33.62565754	8.780053841	23.38691302
331	40.50464092	39.44524976	33.55104949	33.73938852
332	1.480487295	41.52079935	13.52596919	31.67404293
340	27.07799373	35.47286027	22.78862534	25.89629902
341	36.19535766	40.29443531	19.52424012	34.30415317
342	17.41130088	40.41067141	19.15680528	30.70646907
420	37.9988755	40.3800976	0.3348666162	30.78533337
421	20.62649807	37.39540165	10.6953254	29.17010608
422	31.65333826	30.84197019	18.79852813	20.06422332
423	40.43873271	40.48387876	32.5957991	34.34003564
430	13.06604158	31.7095054	21.26088618	21.5224927
431	41.9004134	40.22451736	33.62936613	34.36880833
432	-18.15241417	40.77420378	7.384815728	30.94665914
433	30.7188936	36.78752744	19.94673009	27.91308623
440	9.592825324	31.52137238	12.35097054	20.66246136
441	17.72284948	37.10777187	13.99838878	29.54292822
442	30.79662238	41.02706069	27.07601966	34.5767614
443	13.60292946	40.18054372	12.75177003	31.02908435

Tabla 6.3: Skills para los modelos de PCA al igual que el de referencia (base) de Rincón (2018) para los horizontes de predicción 10 y 30 segundos. La nomenclatura del modelo se corresponde en orden de las cifras: número de clases, número de componentes de PCA, número de la clase (la numeración de la clase empieza en 0 en lugar de en 1)

Modelo	1 minuto	1 minuto	2 minutos	2 minutos
	pca	base	pca	base
220	32.1103556	31.27582573	27.07574122	27.55794299
221	34.48369359	34.24106698	32.78151398	32.73841069
230	32.0148233	31.23998472	27.24490126	27.52506479
231	35.39699304	34.27859981	31.3672137	32.76174755
240	26.47028969	31.20078137	20.51626132	27.50976186
241	29.75860601	34.31833636	11.54783477	32.75971436
320	23.1630781	33.11319613	-3.315657846	32.6569099
321	21.7679013	26.89660523	21.46460068	24.86902359
322	11.1901257	36.6929907	26.85105694	32.072275
330	5.082037142	26.32622304	19.24184415	24.43532361
331	36.33807579	36.55552213	30.70653234	31.98170354
332	19.13527839	33.28258712	-10.94251427	32.71269909
340	26.53669341	28.75574007	22.8413558	25.85525766
341	18.89089507	37.0155079	9.853531014	32.99330675
342	9.096374375	31.66303006	2.255054572	31.93216019
420	21.29697113	31.68352837	18.53097133	31.91276625
421	21.21936977	32.50092589	6.924290976	28.50336394
422	23.43784315	22.85111895	22.68991157	23.15250497
423	35.49081424	37.02864224	30.06004905	33.06868902
430	25.36623839	26.01371173	21.73739706	25.54343544
431	34.7560894	37.16693382	30.07599462	32.97928934
432	21.26291737	31.90786187	3.081397152	32.08219478
433	15.6343089	30.29551067	17.62845028	26.41314504
440	10.74131788	23.88017207	11.77270144	23.78678788
441	8.134106835	33.18857984	-3.695426708	28.80216686
442	30.09421042	36.62947294	13.68946989	32.55171601
443	19.0942252	32.3209597	22.02809975	32.18270431

Tabla 6.4: Skills para los modelos de PCA al igual que el de referencia (base) de Rincón (2018) para los horizontes de predicción 1 y 2 minutos. La nomenclatura del modelo se corresponde en orden de las cifras: número de clases, número de componentes de PCA, número de la clase (la numeración de la clase empieza en 0 en lugar de en 1)

Modelo	5 minutos	5 minutos	10minutos	10minutos
	pca	base	pca	base
220	19.86865958	19.59506296	19.17228849	19.78851143
221	22.22137561	22.83470468	20.77689687	20.6017831
230	19.69630869	19.58861688	19.21193528	19.78902296
231	22.15666843	22.83139681	20.52620507	20.59870727
240	16.59157937	19.58279945	13.44223162	19.78975929
241	8.907118815	22.82684403	17.69648397	20.59515377
320	5.190297001	22.97804492	0.4360988444	20.8526309
321	15.28878961	18.73364152	13.34131269	19.7403408
322	12.79124535	21.55189435	6.769995262	19.76344677
330	7.258461265	18.66594629	1.034603139	19.74868032
331	21.22059276	21.34444242	19.93476391	19.74474501
332	12.90809599	23.01082013	-19.45062282	20.81978711
340	10.07214965	19.03189557	13.81797684	19.73725908
341	-21.2012101	22.26059017	-39.93473756	20.05638513
342	12.77201776	22.70066143	6.591184427	20.80283446
420	12.93777001	22.72303802	16.86166683	20.8331487
421	-78.15357627	18.36175167	-18.316598	17.55435576
422	19.62076724	19.99217276	21.46532731	21.92858047
423	14.92333613	22.35822458	19.73704458	20.09906156
430	19.44828575	22.00540952	21.9645559	24.02081708
431	21.82598616	22.18228271	20.18058965	19.93575631
432	-91.41012569	22.87641334	-17.51044207	20.93846768
433	2.407643273	16.99442933	-25.48370279	16.55872805
440	5.836347788	19.91004235	12.46764536	21.68305273
441	-70.89022562	18.3005216	-139.8832185	17.50496617
442	14.87543137	22.400635	2.017186066	20.2631943
443	2.354056486	22.69442574	14.63999677	20.69669325

Tabla 6.5: Skills para los modelos de PCA al igual que el de referencia (base) de Rincón (2018) para los horizontes de predicción 5 y 10 minutos. La nomenclatura del modelo se corresponde en orden de las cifras: número de clases, número de componentes de PCA, número de la clase (la numeración de la clase empieza en 0 en lugar de en 1)

Capítulo 7

Conclusiones y Trabajo Futuro

7.1. Conclusiones

Los modelos específicos resultan prometedores como alternativa a uno genérico. Como se ha visto en algunos de ellos, donde a pesar de no haber mejorado la calidad de la predicción, por el simple hecho de haber conseguido un resultado muy cercano con una cantidad inferior de datos, hacen uso de menos recursos de tiempo y espacio, que también se traduce en menor coste energético, objetivos que persiguen los entornos IoT.

La exploración llevada a cabo en este estudio ha sido en cierto modo arbitraria como a la hora de elegir el número de clases y de dimensiones para aplicar PCA.

Destacar en el caso de la correlación cruzada que finalmente se han utilizado entre siete y ocho veces las muestras de cada estación respecto al modelo de referencia al haber decidido incluir el máximo de información relevante como indica la desviación estándar entre otras métricas. Esto ha elevado mucho las dimensiones del problema, con lo que la red neuronal ha podido no generalizar en las mejores condiciones para obtener un buen modelo.

Independientemente de la forma de elegir los modelos, al tener algunos con resultados aceptables tanto a partir de la correlación cruzada como de la clasificación y PCA, ambas maneras de definir los subconjuntos de datos se prueban válidos y no causantes de mejoras o empeoramiento de la predicción.

7.2. Trabajo Futuro

En base a este estudio se puede explorar con mayor profundidad las causas que llevan a obtener en algunos modelos skills negativos.

Se propone como línea de trabajo afinar los criterios a partir de la correlación para coger las muestras de cada estación.

A partir de este estudio se puede explorar más en detalle y dirigido la

implementación de PCA y clustering.

Chapter 8

Introduction

8.1. Grounds

Nowadays, in almost every human activity energy is a critical component. Throughout History, many sources have been explored, like fossil fuels which have an immense negative impact on the environment as well as a limited and finite amount is available. Therefore, there is an increasingly and imperative need of research in new energy sources and their usage optimisation which are both enviromentally friendly and more abundant. Those energy sources are categorized as renewable energy like tidal power (using the waves of the great water bodies), wind energy (making use of the winds on the planet surface) or solar energy to mention some examples. Among all of these sources, solar energy occupies this thesis.

Solar irradiance is transformed into energy through solar panels. The level of irradiance fluctuates naturally. The simplest scenario consists of an exact location where the intensity is bound to be different at different times of the day, expecting to be highest at midday or noon. As a consequence, the produced energy is going to vary accordingly. It is of high interest to have some tool or mechanism allowing to know the amount of energy which is going to be produced in the short and medium term as accurately as possible. One application of this would be to determine whether this quantity is going to be enough to cover the needs or should be necessary to reach for other sources like concentration photovoltaics plants or others in order to provide the required amount.

The main purpose consists on predicting the amount of energy to be produced in the short term from a solar power plant based on the solar irradiance. This forecast is going to rely upon previous data measurements. To this end, there are several techniques whith different approaches like using satellite imagery of the area or taking and analysing images of the sky from the groud to mention some of them. An Internet of Thing (IoT) system brings about another way of collecting the necessary information using a

sensor network. In order to do this, several nodes are deployed throughout the area to take measurements of solar radiation as well as other parameters like temperature or humidity at a given frequency. Note that deployment is not covered by this work. All the registered data is going to be sent to a device of higher computing capacity for analysing and predicting using statistic models. Thus, creating an IoT environment.

8.2. Objectives

There are previous works on machine learning to obtain models whose purpose is predicting the energy to be produced like Rincón (2018). The learning process needs a training on all data considered pertinent.

According to the criteria for selecting the data, one inital set of it may derive in several distinct models. Among this criteria there is the selection of measurement points to be used (which may differ from the total available), or the number of samples for each node, or time intervals like the whole year, a single season, months, et cetera. All these possibbilities derives on a great amount of more or less specialised models. Those more generic may present some noise from the data in addition to increase the computation requirements in both time and memory space provided it has to process more information than the amount strictly needed.

The purpose of this thesis is explore the effect on the prediction accuracy when several models are being used, covering each one a particular scenario, instead of a single model for everything. The underlying hypothesis is to increase the accuracy of the prediction based on the relationship and affinity among the measurements from different nodes of the network. This determines the stations as well as the amount of data for each one being relevant regarding the forecasting at a previously established location during some periods of time (inferiors to the whole available).

8.3. Work plan

On one hand, there is a study using cross correlation between the whole data set of all the measurement stations to find their affinity and select the subsets of data accordingly. Solar irradiation is affected by weather variations like winds carrying clouds covering momentarily the solar panel reducing in turn the amount of received irradiation. As a consequence, in addition to applying cross correlation to the whole data, some time based scenarios are tested such as seasons, months or a period of time of some hours like mornings.

On the other hand, as an additional implementation of the aforementioned where the different models are defined almos manually, there is a

study relying on the principal component analysis (PCA) technique and unsupervised clustering in several classes. Each class is going to be a model. This unsupervised machine learning allows evaluating the first method of model selection.

Chapter 9

Conclusions and Future Work

9.1. Conclusions

As an alternative to the global model, some more specific models may be used. Even if the best results of these new models are not better than the reference model, they are highly close by using a subset of the original data. Therefore, they behave in a similar fashion regarding prediction with less resources in time and space resulting in a lower power consumption which is essential in an IoT environment. Since by definition IoT implies low resource requirements aong other aspects.

In order to conduct this study some decisions were arbitrarily taken, since no information to base upon this was found like the number of clusters and PCA dimensions. In order to reduce its impact, we have a small exploration covering the first set of combinations of these variables.

Regarding the cross correlation, it is important to highlight the fact of using between seven and eight times the number of samples for each station as a result of trying to include the maximum relavant information as stated by metrics like the standard deviation. As a consequence, instead of reducing the number of nodes, even excluding some of them, the number of dimensions to be treated is heavily increased leading to far than better conditions for the neural network to obtain a good model.

As some models stemming from both the manual selection guided by the cross correlation on one hand and from clustering and PCA on the other hand have acceptable results, both ways of model definition are proven valid and do not create neither improvement nor degradation in predciting solar irradiance.

9.2. Future work

Stemming from this study, a deeper exploration to understand the causes behing negative skill may be engaged.

As a future line of work, it seems relevant to improve the criteria to apply on the cross correlation results for obtaining the number of samples of each station.

This study may be used as a startig point to research on the best implementation for PCA and clustering in a slightly more directed way as well as more detailed.

Bibliografía

Bourguignat, C. A tensorflow tutorial. Disponible en https://github.com/christophebourguignat/notebooks/blob/master/TensorFlow%20Tutorial.ipynb.

Cebreiros, M. M. *Diseño e Implementación de un Nodo Sensor de Radiación*. Proyecto Fin de Carrera, 2017.

scikit-learn developers. Installing scikit-learn. Disponible en https://scikit-learn.org/stable/install.html.

Eschenbach, A. Short-term solar irradiation from a sparse pyranometer network. 2018.

Firing, E. *Oceanographic Data Analysis With Open Source Tools*. Versión electrónica, 2019. Disponible en https://currents.soest.hawaii.edu/ocn_data_analysis/_static/SEM_EDOF.html#Cross-correlationl (último acceso, Septiembre 2019).

Jia, Y. Interfaces. Disponible en https://caffe.berkeleyvision.org/tutorial/interfaces.html.

Jia, Y. Layers. Disponible en https://caffe.berkeleyvision.org/tutorial/layers.html.

scikit learn, D. Multi-layer perceptron. Disponible en https://github.com/scikit-learn/scikit-learn/blob/7389dba/sklearn/neural_network/multilayer_perceptron.py#L1065.

Maxim. tensorflow neural network multi layer perceptron for regression example. Disponible en https://stackoverflow.com/questions/46832151/tensorflow-neural-network-multi-layer-perceptron-for-regression-example.

Palomino, I. G. y Pérez, F. J. H. Desarrollo de red de sensores de irradiación solar. 2019.

Rincón, G. Y. *Infraestructura para la predicción de radiación solar a corto plazo*. Proyecto Fin de Carrera, 2018.

KERAS TEAM. Keras: The python deep learning library. Disponible en `https://keras.io/#installation`.

KERAS TEAM. The sequential model api. Disponible en `https://keras.io/models/sequential/`.

TENSORFLOW, D. Install tensorflow. Disponible en `https://www.tensorflow.org/install`.

TENSORFLOW, D. tf.estimator.dnnregressor. Disponible en `https://www.tensorflow.org/api_docs/python/tf/estimator/DNNRegressor`.

VANDERPLAS, J. *Python Data Science Handbook*. Versión electrónica, 2016. Disponible en `https://jakevdp.github.io/PythonDataScienceHandbook/05.09-principal-component-analysis.html` (último acceso, Septiembre 2019).