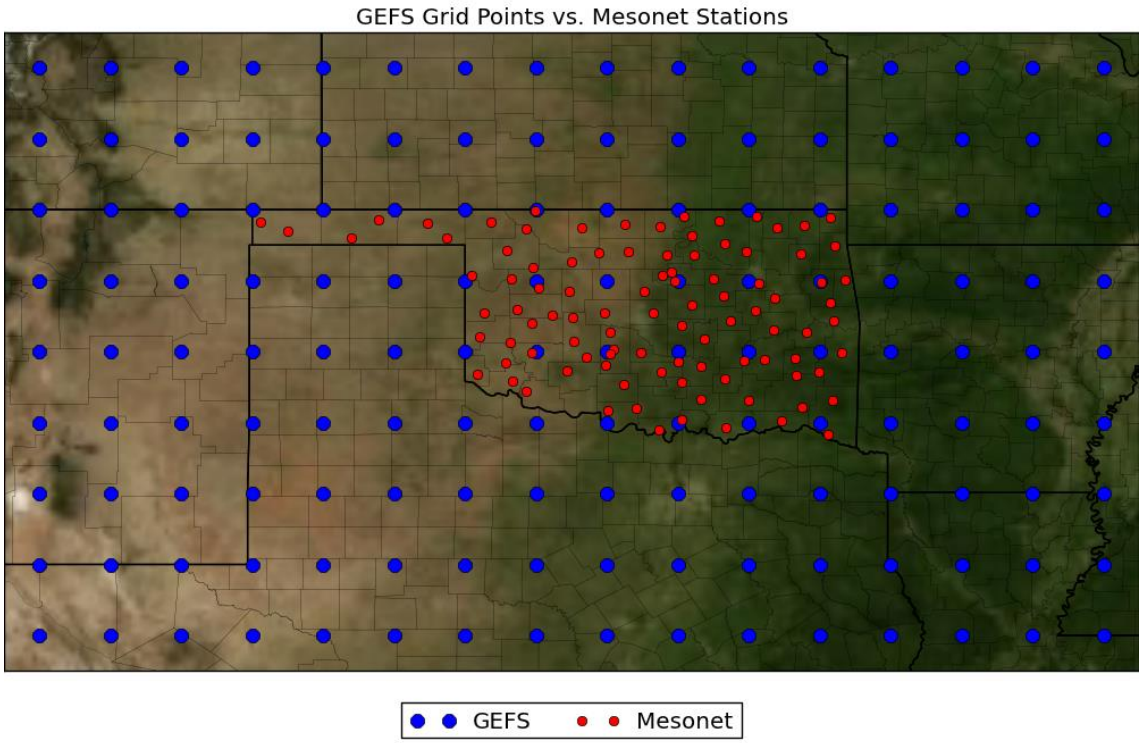
PRÁCTICA 1: REGRESIÓN CON MLR3 (4 puntos)

APLICACIÓN DEL APRENDIZAJE AUTOMÁTICO EN LA INDUSTRIA ENERGÉTICA: PREDICCIÓN DE RADIACIÓN SOLAR EN PLANTAS FOTOVOLTAICAS



Se trata de crear modelos de regresión para resolver un problema de predicción de radiación solar a partir de predicciones de variables meteorológicas. En la figura se puede ver un gráfico con puntos rojos y azules. Los puntos rojos son plantas solares (donde se convierte energía solar en energía eléctrica). Los puntos azules son lugares donde simulaciones matemáticas (conocidas como modelos NWP = Numerical Weather Prediction) realizan predicciones de algunas variables meteorológicas, tales como apcp_sfc (precipitación acumulada durante tres horas), o dswrf_sfc (radiación de onda corta en la superficie). Se pueden ver los nombres y significado de las variables en https://www.kaggle.com/c/ams-2014-solar-energy-prediction-contest/data. Sin embargo, hay que tener en cuenta que el conjunto de datos no es el original sino que ha sido modificado para la práctica.

NOTAS IMPORTANTES:

1) Los resultados tienen que ser reproducibles. Como sabéis, para ello tenéis que usar set.seed() en los lugares apropiados. Pero en lugar de usar set.seed(0), tenéis que usar set.seed con el NIA de uno de los miembros del grupo. Ejemplo, si vuestro NIA es 100345719, entonces tenéis que poner set.seed(100345719).
2) Aunque el problema es el mismo para todos (predicción de radiación), cada grupo va a trabajar con una estación (punto rojo) distinto. El identificador de la estación de cada grupo se recoge en el fichero adjunto "NÚMERO DE FICHERO DE DATOS A USAR".
3) Se suministran dos ficheros de datos:
    a. Un primer fichero de datos disponibles, llamado "disp_xx.rds", para conseguir un modelo final que sea el mejor posible, y su estimación. Cada instancia corresponde a 1 día del año, y están ordenados por día. Este primer fichero tiene 75 atributos de entrada. La variable de respuesta se llama "salida".
    b. Un segundo fichero de datos para hacer predicciones con el modelo final, llamado "compet_xx.rds". Este segundo fichero no tiene la variable de respuesta (puesto que es desconocida y tiene que ser predicha con el modelo final). .rds es un formato especial de R, que permite guardar datos ocupando menos espacio. Se cargan en Rstudio así: datos_disp = readRDS('disp_1.rds').
4) Cada uno de los apartados tiene su puntuación máxima, que se obtendrá si el código MLR es correcto y si hay un comentario claro de los resultados en la memoria.

SE PIDE:

1. **(0.25 puntos) Exploratory Data Análisis:** Hacer un EDA (Exploratory Data Analysis) sencillo (No DataExplorer). Se trata de saber:
    a. Cuántos datos (instancias) y atributos hay.
    b. De qué tipo son los atributos.
    c. Si tienen missing values (NA's) y qué proporción de sus valores son NA's.
    d. Si hay atributos constantes.
    e. Un plot de la variable de respuesta a lo largo del tiempo (y su interpretación).

2. **(0.25 puntos) Métrica: relative absolute error (RAE):** Esta va a ser la métrica que vamos a usar para evaluar los modelos. Describidla usando vuestras propias palabras y encontrad cómo usarla en mlr3.

3. **(3 puntos) Determinar cuál es el mejor método para estos datos**

    Recordad que en esta sección, sólo se pueden usar los datos de los 9 primeros años (desde el dato 1 hasta el dato 9*365). Los 3 últimos años serán para test. *Custom resample* no funciona con AutoTunner para evaluar en las comparaciones del apartado, usad el código que se adjunta en Aula Global.

2.1. (0.25 puntos) ¿Cuál es el mejor método de imputación y de escalado?

Se trata de comparar varios métodos de imputación y de escalado, y determinar cuál es la mejor opción. Lo haremos sólo para el vecino más cercano con hiper-parámetros por omisión (default). En este apartado, me gustaría ver dos tablas, una donde se evalúan distintas maneras de imputar, y otra donde se evalúan las distintas maneras de escalar (normalizar). Por evaluar me refiero a calcular el RAE en el conjunto de validación.

2.2. (0.25 Puntos) Evaluar varios métodos SIN ajuste de hiper-parámetros (o sea, con hiper-parámetros por omisión / default)

Este apartado se hará con los siguientes métodos: regr.*lm*, *rpart*, vecino más cercano, *cubist*, y *SVM* lineal y radial (kernel gausiano).

Usad una tabla para mostrar los resultados. Sacad algunas conclusiones generales: por ejemplo, ¿cuál es el mejor método?, ¿se trata de un problema donde los métodos no-lineales funcionan mucho mejor que los lineales?.

2.3. (1 punto) Ahora vamos a hacer lo mismo, pero con ajuste de hiper-parámetros:

a. Para el vecino más cercano, ajustad sólo el número de vecinos con *Grid Search.*
b. Para *rpart* y SVM usad *Random Search.*
c. De *cubist* **no** vamos a hacer ajuste de hiper-parámetros.
d. De regr.lm **no** vamos a hacer ajuste de hiper-parámetros.

Para cada método, ¿se consigue mejorar los resultados? ¿Se consigue ahora un mejor resultado global que el que se obtenía en el punto anterior?.

2.4. (0.5 puntos) Ahora probad con métodos de ensembles SIN Y CON ajuste de hiper-parámetros:

e. Random Forest: con *ranger*. Ajustad sólo sus dos hiper-parámetros más importantes.
f. Gradient Boosting: podéis elegir uno de dos métodos *catboost* o *xgboost*. Ajustad sólo sus dos hiper-parámetros más importantes.

¿Se mejoran los resultados hasta el momento? ¿Aporta algo el ajuste de hiper-parámetros para los ensembles en este problema?.

4. **(1 punto) Basándose en todos los resultados de la práctica, elegid el mejor método para entrenar el modelo final y computad su estimación. Es decir, hay que hacer lo siguiente:**
    a. Obtener la estimación del error del modelo final hay que hacerla con los datos que habíamos reservado para ello. O sea, usando como datos de test los tres últimos años.
    b. Obtener el propio modelo final. Este modelo hay que guardarlo en un fichero y entregarlo.
    c. Y usar el modelo final para computar las predicciones sobre el conjunto compet_xx.rds, las cuales las guardaréis en un fichero de texto, que también se entregarán.
5. **(0,5 puntos)** Este último punto es para que investiguéis vosotros. Se trata de encontrar información sobre el método de ajuste de hiper-parámetros denominado “hyperband”,

describidlo en la memoria usando vuestras palabras, y usadlo para ajustar hiper-parámetros de un método, el que elijáis, describiendo las ventajas que se observan al usarlo.

A ENTREGAR EN UN FICHERO ZIP A TRAVÉS DE AULA GLOBAL:

- Uno o varios ficheros con el código R. Puede ser el script, o bien ficheros .Rmd (R markdown).
- Una memoria. Puede ser uno o varios .pdf, .doc, .html, o también el resultado de ejecutar el .Rmd (en .html, .doc, o .pdf).
- Un fichero de texto con las predicciones para los datos de test con el modelo final.
- El modelo final, guardado con saveRDS.
- Un video de no más de 5 minutos exponiendo las conclusiones de la memoria y vuestra valoración sobre la práctica.