

## **Treball final de grau**

**Estudi:** Grau en Enginyeria Informàtica

**Títol:** Mòdul de predicció per al consum energètic

**Document:** Resum

**Alumne:** Àlex López Diaz

**Tutor:** Llorenç Burgas

**Departament:** EEIA

**Àrea:** ENGINYERIA DE SISTEMES I AUTOMÀTICA

**Convocatòria (mes/any):** Setembre/2021

## Introducció i objectius

El *forecasting* de sèries de temporals basat en tècniques d'intel·ligència artificial és una línia de recerca d'interès dins de diversos sectors, un d'ells és la demanda energètica d'edificis. Ja que si som capaços de preveure quin consum tindrem, serem capaços d'actuar en l'emmagatzematge d'energia de forma més òptima. A *l'estat de l'art* existeixen moltes tècniques i metodologies diferents per realitzar *forecastings*.

L'objectiu d'aquest treball és elaborar una eina de *forecasting* autònoma que permeti entrenar els models de manera que sigui el màxim autònoma possible, emmagatzemar-los i posteriorment utilitzar-los per obtenir previsions d'energia consumida. En aquest treball s'implementarà un mòdul que s'integrarà dins la plataforma software del projecte europeu e-Land.

En aquest treball es provaran i utilitzaran diverses tècniques d'intel·ligència artificial per aconseguir els millors resultats de *forecasting* possibles. S'utilitzaran tècniques d'autoselecció de paràmetres per als diversos algorismes de *forecasting* seleccionats i es gestionarà la possibilitat de tenir diversos models / usuaris i el seu emmagatzematge. Finalment es testejarà la metodologia implementada en un cas d'ús utilitzant dades reals d'un dels 3 pilots del projecte e-Land.

## Metodologia i planificació

He seguit la metodologia SCRUM i els *sprints* han quedat així:

- Sprint 1. El primer *sprint* és dedicat a investigar i aprendre sobre la regressió i aplicacions relacionades amb les sèries temporals..
- Sprint 2. Estudi i modificació de les dades d'entrada al model.
- Sprint 3. Investigació i decisió sobre els models predictors aplicats a les sèries temporals.
- Sprint 4. Muntatge de l'estructura de dades dins els models. Com emmagatzemem les dades a cada model.

- Sprint 5. Represa del projecte; veure què està fer i implementar el primer model predictor del sistema, SVR.
- Sprint 6. Implementació del SARIMAX.
- Sprint 7. Proves dels dos models implementats. Al veure els resultats es planteja l'opció afegir un nou model al sistema.
- Sprint 8. Implementació LSTM.
- Sprint 9. Simulació d'un cas d'ús real on entrem dades periòdicament al nostre sistema.
- Sprint 10. Redacció de la memòria.

## Requisits, estudis i decisions

L'eina ha de proporcionar les funcionalitats necessàries com per poder predir el consum energètic d'un edifici, rebent dades periòdicament, guardant la predicció obtinguda i tenint una mesura que permeti que es re-calculi de forma automàtica.

En quant a decisions, per realitzar aquest projecte s'ha d'escollit un llenguatge versàtil y potent en càlculs estadístics i amb gestió de dades incorporat: Python.

S'ha escollit aquest llenguatge per les llibreries que ofereix. *Sklearn i Tensorflow* en son les més conegudes.

Per dur a terme el treball s'ha de determinar quins algorismes predictors s'apliquen. Es decideix aplicar els següents algorismes:

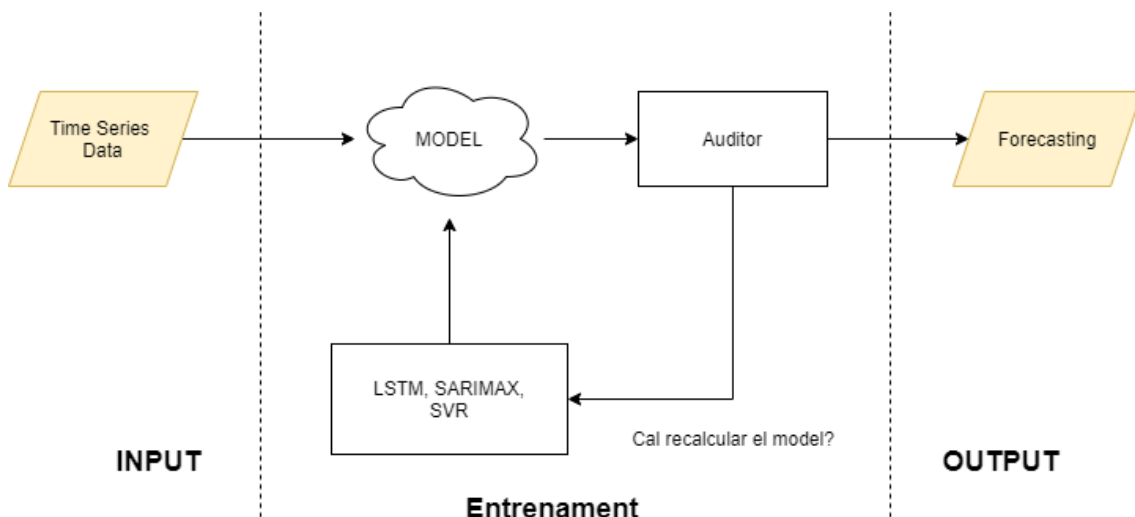
- **ARIMA.** Acrònim de l'anglès "*autoregressive integrated moving average*", és un model estadístic específic per predir sèries de dades temporals.
- **SVM.** Són un conjunt de mètodes d'aprenentatge supervisat utilitzats per classificació de dades, regressió i detecció de valors atípics.
- **Xarxa neuronal.** Una xarxa neuronal és un sistema informàtic inspirat en les xarxes neuronals biològiques que construeixen el cervell animal.
- **Model per mitjanes.** Consta de treure la mitjana del dia a predir de totes les dades guardades en el model.

## Anàlisi, disseny i implementació

La proposta per aquest projecte pel problema de la gestió del consum energètic és d'un sistema on tractarem cada dia de la setmana per separat.

Primerament obtindrem una sèrie temporal de dades com a paràmetre d'entrada al sistema, de la qual generarem un primer model (dividint les dades per dies) i crearà la primera predicció del sistema. A continuació, entra en joc una entitat (auditor) que determinarà si s'ha de re-calcular el model o no, depenent de l'error que obtingui entre les noves dades entrades al model (dades reals) i la predicció generada pel model.

Un cop supera l'error o ens quedem sense dades predites que hem guardat en el model, es re-calcula el model, aplicant tots els algorismes esmentats: LSTM, SARIMAX i SVR.



Per comprovar els resultats es duran a terme diverses proves, tant a les prediccions fetes al model com als paràmetres de cada algorisme per treure'n resultats competents.

## Resultats i conclusions

El resultat del nostre model ve determinat per l'eficiència dels mòduls SARIMAX, SVR i LSTM. S'executen tots els algorismes cada vegada que es vol calcular un model.

Els resultats obtinguts a llarg termini són:

- Precisió SVR: 85.18%
- Precisió SARIMAX: 58.34%
- Precisió LSTM: 81.20%
- Precisió Mitjanes: 74.54%

En general, els resultats obtinguts són bons. Els resultats més destacables són els de l'SVR i l'LSTM, que competeixen en la majoria de prediccions.

Com a treball futur, quedaria implementar algun altre algorisme per competir amb la resta, ja que el codi és molt modular i fàcilment adaptable. També considero que aprofundir en l'àmbit del *Deep Learning* seria molt interessant en relació a l'objectiu del projecte.

Per últim, quedaria provar l'eina implementada en un servidor i de forma totalment automàtica on s'entren les dades periòdicament.