

Simulación de Sistemas Eléctricos

Actividad 7: Modelo de predicción

- **Variable independiente (x):** Energía en $t=t_0$ (Valor actual)
- **Variable dependiente (f(x)):** Energía en $t=t_0+dt$ (Valor siguiente); $dt=1\text{seg}$
- **Frecuencia de muestreo de los datos:** 1 muestra por segundo (sps)
- **Venta temporal de predicción:** 1 segundo ($dt=1\text{seg}$)

Nombres:

- Estudiante 1: VA
- Estudiante 2: VA

Contenido:

- Descargar datos (.csv) desde repositorio (DataPort)
- **Paper:** <https://doi.org/10.1016/j.procs.2022.07.035>
- **Dataset:** <https://ieee-dataport.org/open-access/data-server-energy-consumption-dataset>
- **Codigo Matlab:** <https://github.com/vasanza/EnergyConsumptionPrediction>
- **Funciones de Matlab:** https://github.com/vasanza/Matlab_Code
- **1.- Leer el archivo .csv usando Matlab y graficar los datos**
- **2.- Adecuar el dataset con la var. independiente (x) y la var. dependiente f(x)**
- **3.- Cambiar la frecuencia de muestreo de los datos (1 muestra por dia)**
- **4.- Toolbox de Matlab - Regression Learner (24 algoritmos)**
- **5.- Seleccionar el modelo que tiene un menor error de predicción**
- **6.- Crear la función del mejor modelo**
- **Nota:** realizar cualquier cambio que usted considere adecuado
- **Despues:**
- Utilizar más variables independientes
- Cambiar la ventana temporal de predicción ($dt > 1\text{seg}$)

Eliminar datos de memoria y command window

```
clear; %Borrar el kspace  
clc; %Borrar el command window
```

Configuración de carpeta /src para librerías

```
%nombre de la carpeta donde estan los codigos  
addpath(genpath('./src'));
```

Configuración de carpeta de /data para datasets

```
%Nombre de la carpeta donde estan los archivos csv  
datapath=fullfile('./data/');
```

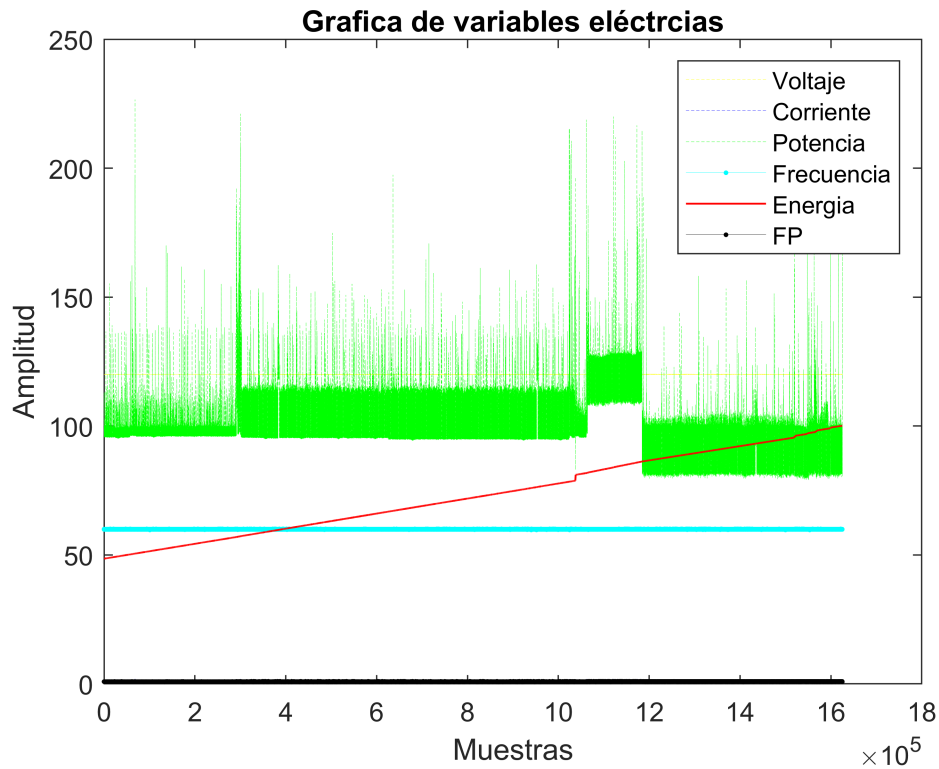
```
filenames=FindCSV(datapath);
```

1.- Leer el archivo .csv usando Matlab y graficar los datos

```
%Esto es lo que se hace con datos .csv  
data=readtable(fullfile(datapath,filenames(1).name));  
%clear dataRaw; %Borrar de memoria la variable que no usare despues  
%Seleccionar las variables electricas de interes  
data2=table2array(data(:,1:6)); % Convertir datos de string a float o int  
data2(isnan(data2))=0; % Eliminar los datos NaN
```

Graficar todas las Variables

```
Voltaje=data2(:,1);  
Corriente=data2(:,2);  
Potencia=data2(:,3);  
Frecuencia=data2(:,4);  
Energia=data2(:,5);  
FP=data2(:,6);  
  
figure %crear una nueva grafica  
%plot(data2) %plotea la grafica 1  
plot(Voltaje,'y--',LineWidth=0.1);  
hold on %habilita plotear mas de una grafica en la misma figura  
plot(Corriente,'b--',LineWidth=0.1);  
plot(Potencia,'g--',LineWidth=0.01);  
plot(Frecuencia,'c.-',LineWidth=0.1);  
plot(Energia,'r',LineWidth=0.7);  
plot(FP,'k.-',LineWidth=0.1);  
hold off  
  
title('Grafica de variables eléctricas');  
xlabel('Muestras');  
ylabel('Amplitud');  
legend('Voltaje','Corriente','Potencia','Frecuencia','Energia','FP');
```



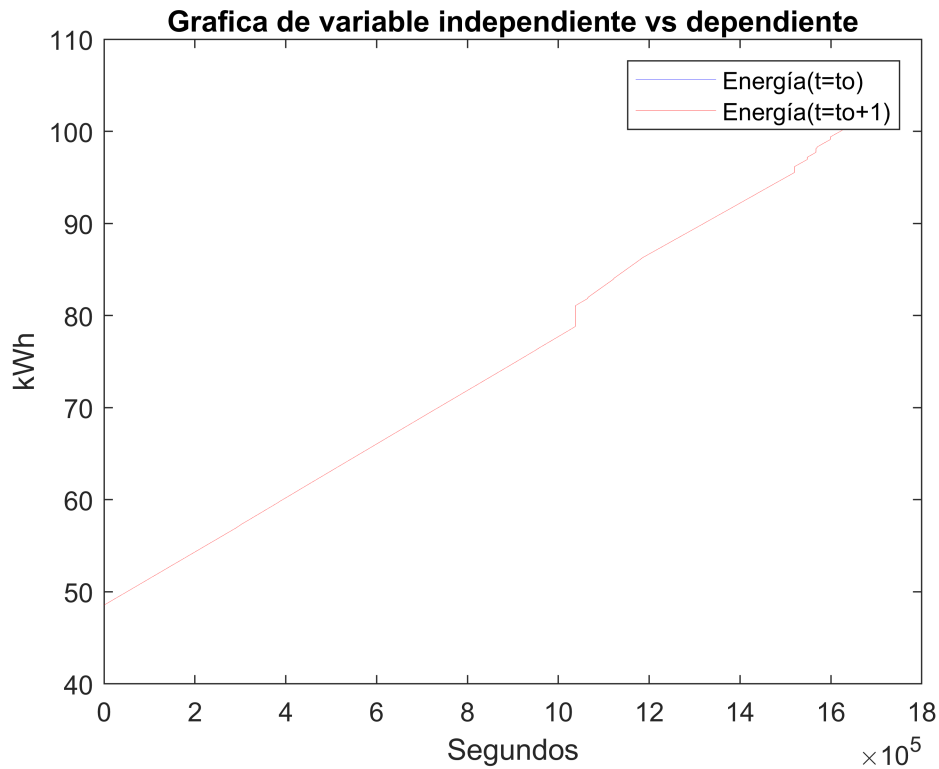
```
%Limpiar del workspace variables no utilizadas
clear FP; clear Potencia; clear Frecuencia; clear Corriente; clear Voltaje;
clear data; clear data2; clear datapath;
```

2.- Adecuador el dataset con la var. independiente (x) y la var. dependiente f(x)

```
input=Energia(1:size(Energia,1)-1,1);% Variable independiente (x)
output=Energia(2:size(Energia,1),1);% Variable dependent f(x)

%Primeras columnas son las entradas o varianles independientes
%La ultima columna es la salida o variable dependiente
dataset=[input output];

figure %crear una nueva grafica
plot(input(:,1),'b',LineWidth=0.1);
hold on %habilita plotear mas de una grafica en la misma figura
plot(output(:,1),'r',LineWidth=0.1);
hold off
%
title('Grafica de variable independiente vs dependiente');
xlabel('Segundos');
ylabel('kWh');
legend('Energía(t=to)', 'Energía(t=to+1)');
```

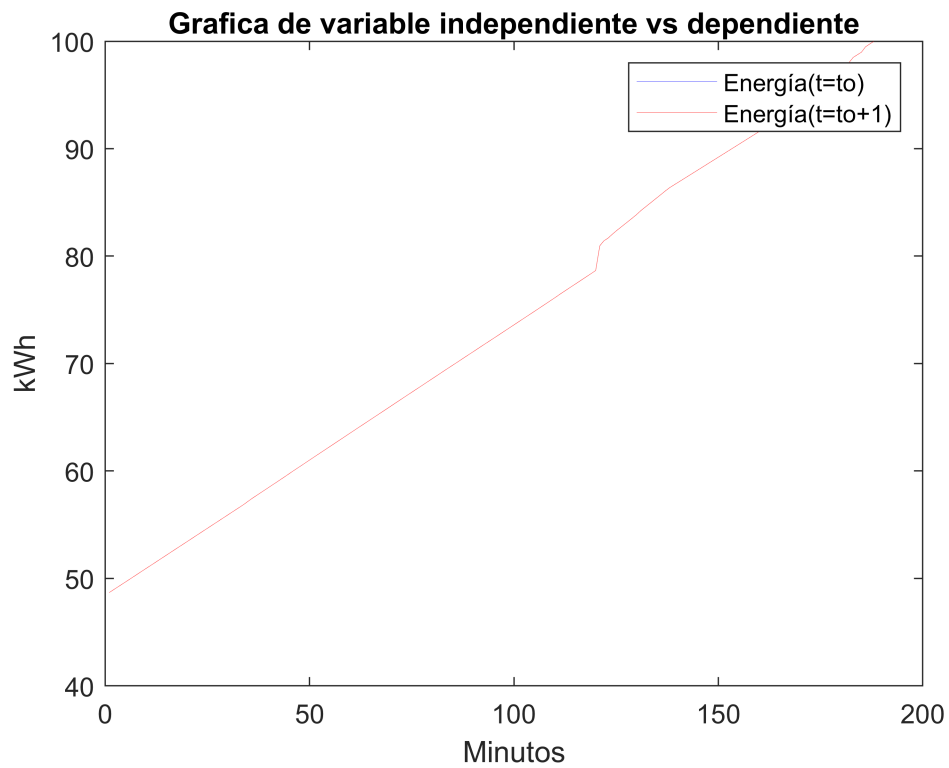


3.- Cambiar la frecuencia de muestreo de los datos (1 muestra por día)

```
win=360*24; %dado a que los datos estan en segundos y queremos llevarlo a dias
datasetMean=fData_MeanWin(dataset,win);%60 segundos

%70 de entrenamiento y 30% de vtesting
Training=datasetMean(1:round(size(datasetMean,1)*0.7),:);% fila 1 hasta la fila del 70% de toda
%desde la fila del 70% de todas las filas hasta la ultima fila
Testing=datasetMean(round(size(datasetMean,1)*0.7)+1:round(size(datasetMean,1)),:);

figure %crear una nueva grafica
plot(datasetMean(:,1),'b',LineWidth=0.1);
hold on %habilita plotear mas de una grafica en la misma figura
plot(datasetMean(:,2),'r',LineWidth=0.1);
hold off
%
title('Grafica de variable independiente vs dependiente');
xlabel('Dias');
ylabel('kWh');
legend('Energía(t=to)', 'Energía(t=to+1)');
```



4.- ToolBox de Matlab - Regression Learner (24 algoritmos)

regressionLearner

New Session from Workspace

Data set

Data Set Variable

dataset1625174x2double

Use columns as variables

Use rows as variables

Response

From data set variable

From workspace

column_2double48.56 ...

Predictors

	Name	Type	Range
<input checked="" type="checkbox"/>	column_1	double	48.56 .. 100.12
<input type="checkbox"/>	column_2	double	48.56 .. 100.12

Add All

Remove All

[How to prepare data](#)

Refresh

Validation

Validation Scheme

Cross-Validation

Protects against overfitting. For data not set aside for testing, the app partitions the data into folds and estimates the accuracy on each fold.

Cross-validation folds:10

[Read about validation](#)

Test

☐ Set aside a test data set

Percent set aside:0

Use a test set to evaluate model performance after tuning and training models. To import a separate test set instead of partitioning the current data set, use the Test Data button after starting an app session.

[Read about test data](#)

Start Session

Cancel

5.- Seleccionar el modelo que tiene un menor error de predicción

Regression Learner - untitled*

FILE

Open

Save

OPTIONS

Feature Selection

PCA

Optimizer

MODELS

All Quick-To-Train

All

Linear

Interactions

Linear

Duplicate

Delete

TRAIN

Use Parallel

Train All

PLOTS

Response

Predicted vs. Actual

Residuals (Validation)

TEST

Layout

Test Data

Test All

EXPORT

Export Plot to Figure

Generate Function

Export Model

Models

Sort by: Model Number

2.1	Linear Regression	RMSE (Validation): 2.0561e-05
Last change: Linear		
2.2	Linear Regression	RMSE (Validation): 2.0561e-05
Last change: Interactions Linear		
2.3	Linear Regression	RMSE (Validation): 2.0545e-05
Last change: Robust Linear		
2.4	Stepwise Linear Regre...	RMSE (Validation): 2.0561e-05
Last change: Stepwise Linear		
2.5	Tree	RMSE (Validation): 0.77867
Last change: Fine Tree		
2.6	Tree	RMSE (Validation): 1.8279
Last change: Medium Tree		
2.7	Tree	RMSE (Validation): 4.1681
Last change: Coarse Tree		
2.8	SVM	RMSE (Validation): 1.1593
Last change: Linear SVM		
2.9	SVM	RMSE (Validation): 1.3673
Last change: Quadratic SVM		
2.10	SVM	RMSE (Validation): 1.4053
Last change: Cubic SVM		
2.11	SVM	RMSE (Validation): 1.5361
Last change: Fine Gaussian SVM		
2.12	SVM	RMSE (Validation): 1.1994
Last change: Medium Gaussian SVM		

Model 2.1

Model 2.3

Summary

Response Plot

Predictions: model 2.3

Plot

True

Predicted

Errors

Style

Markers

Box plot

X-axis

X: Record number

[How to use the response plot](#)

Data set: dataset/Mean

Observations: 188

Size: 4 kB

Predictors: 1

Response: column_2

Validation: 10-fold Cross-Validation

6

6.- Crear la función del mejor modelo

```
[trainedModel, validationRMSE] = trainRegressionModel(Training)
```

```
trainedModel = struct with fields:
```

```
    predictFcn: @(x)linearModelPredictFcn(predictorExtractionFcn(x))
```

```
    LinearModel: [1x1 LinearModel]
```

```
    About: 'This struct is a trained model exported from Regression Learner R2022a.'
```

```
    HowToPredict: 'To make predictions on a new predictor column matrix, X, use: yfit = c.predictFcn(X) ↵replac
```

```
validationRMSE = 2.2664e-05
```

```
EnergiaEst=trainedModel.predictFcn(Testing(:,1));
```

```
figure %crear una nueva grafica
```

```
plot(Testing(:,2),'b',LineWidth=0.1);
```

```
hold on %habilita plotear mas de una grafica en la misma figura
```

```
plot(EnergiaEst,'r',LineWidth=0.1);
```

```
hold off
```

```
%
```

```
title('Valor real vs estimada');
```

```
xlabel('Dias');
```

```
ylabel('kWh');
```

```
legend('Real','Est');
```

