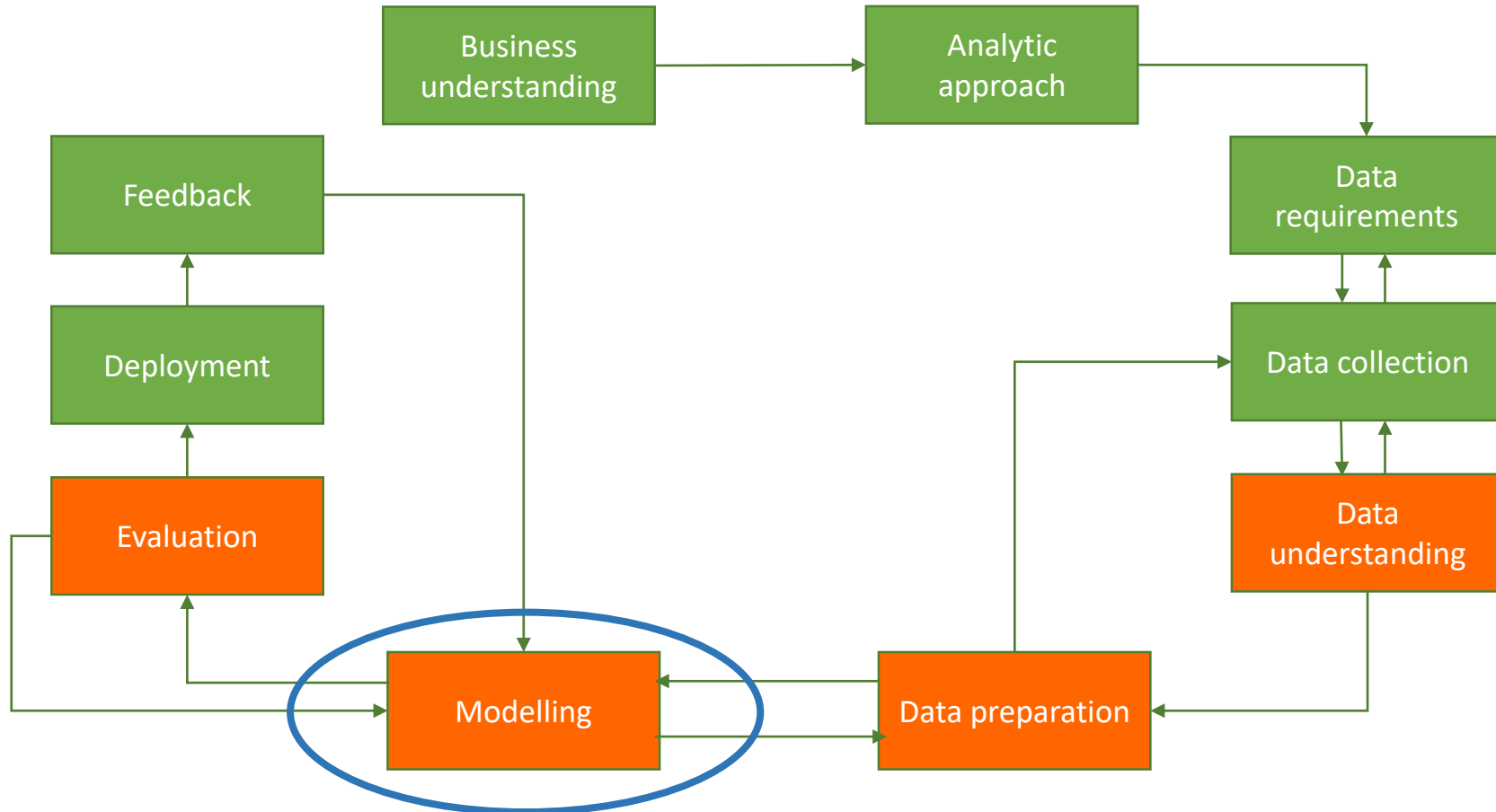


Modelado de datos

Parte 2

Análisis de Datos - Paola A. Sánchez-Sánchez

Metodología para el análisis de datos



Agenda

- Predicción: Modelos de regresion lineal multiple
- Modelos de redes neuronales
- Modelos de regresion en Python



Regresión lineal múltiple

Cuando dos o más variables independientes influyen sobre una variable dependiente

Regresión Lineal Múltiple

$$y = a_1x_1 + a_2x_2 + a_nx_n + b$$

Variable
dependiente

$a_{1..n}$

Coeficientes

$x_{1..n}$

Variables
independientes

Intersección

Supuestos sobre el modelo

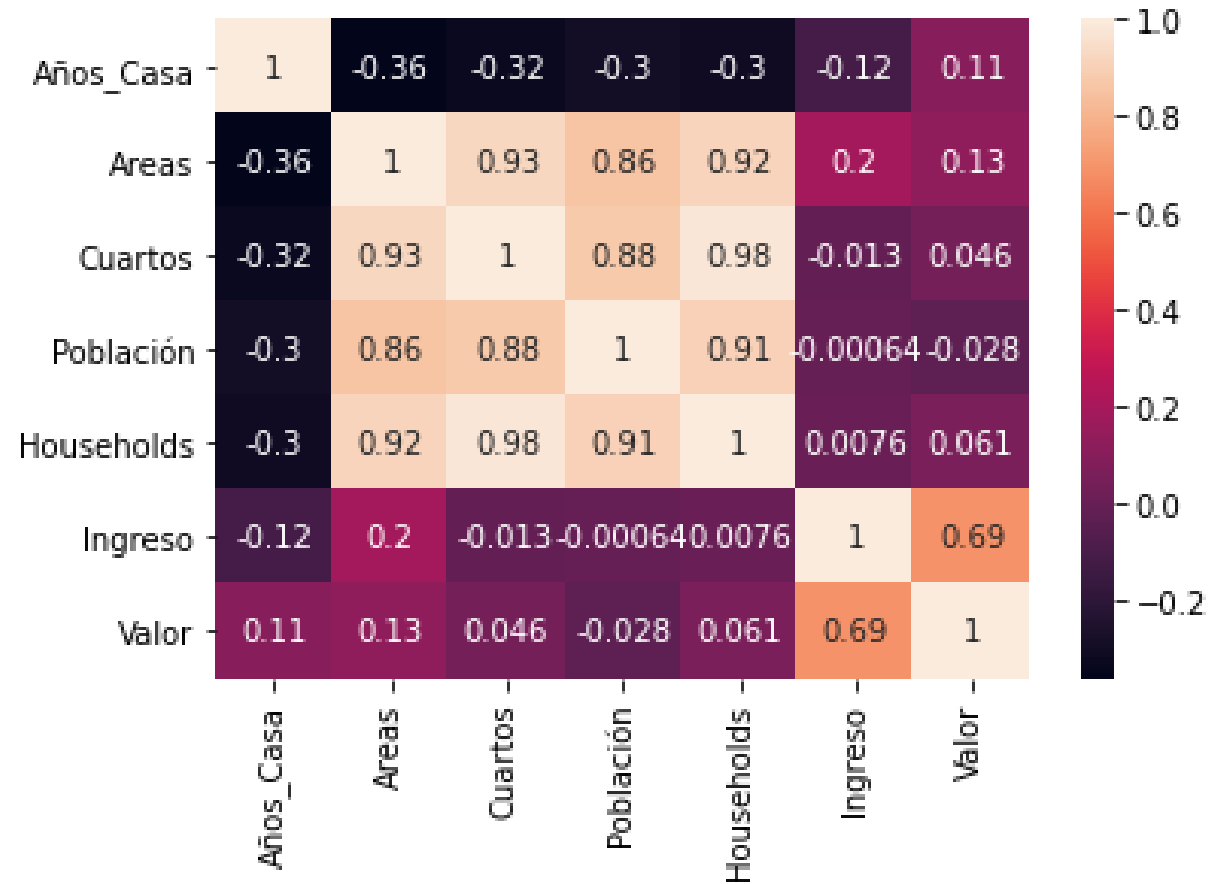
Existe una relación lineal y aditiva, entre las variables dependientes e independientes.

No debe haber correlación entre las variables independientes.

Los términos de error deben poseer varianza constante, ni deben correlacionarse.

La variable dependiente y los términos de error deben tener una distribución normal

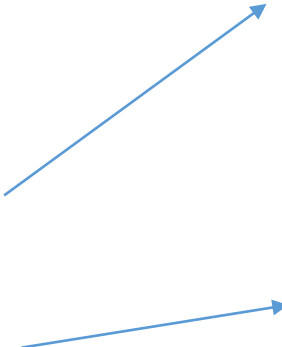
Medir correlación entre variables



Basado en california_housing_train.csv

Implementación de modelo RLM en Python

1. Importar librerías
2. Cargar Datos/selección/renombrar
3. Limpieza de Datos
4. Transformar variables categóricas
5. Correlación
6. Partición de conjunto de datos
7. Ajuste del modelo y predicción
8. Cálculo de parámetros, precisión



```
#Defino el algoritmo a utilizar
lr = linear_model.LinearRegression()

#Entreno el modelo
lr.fit(X_train, y_train)

#Realizo una predicción
Y_pred = lr.predict(X_test)
```

```
#Calculo de coeficiente
a = lr.coef_

#Calculo de intercepto
b = lr.intercept_

#Calculo de precisión R2
R2 = lr.score(X_test, y_test)
```

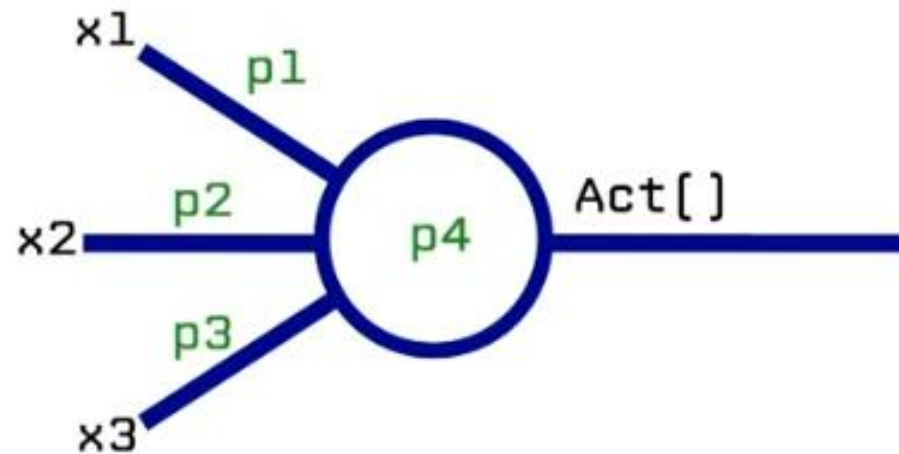
Redes neuronales

QUE SON?

- Tipo de modelo que se usa en ML para aprender de los datos
- Se usa en aprendizaje supervisado y no supervisado
- Se usa en tareas de regresión y clasificación
- Basados en las conexiones de las neuronas del cerebro

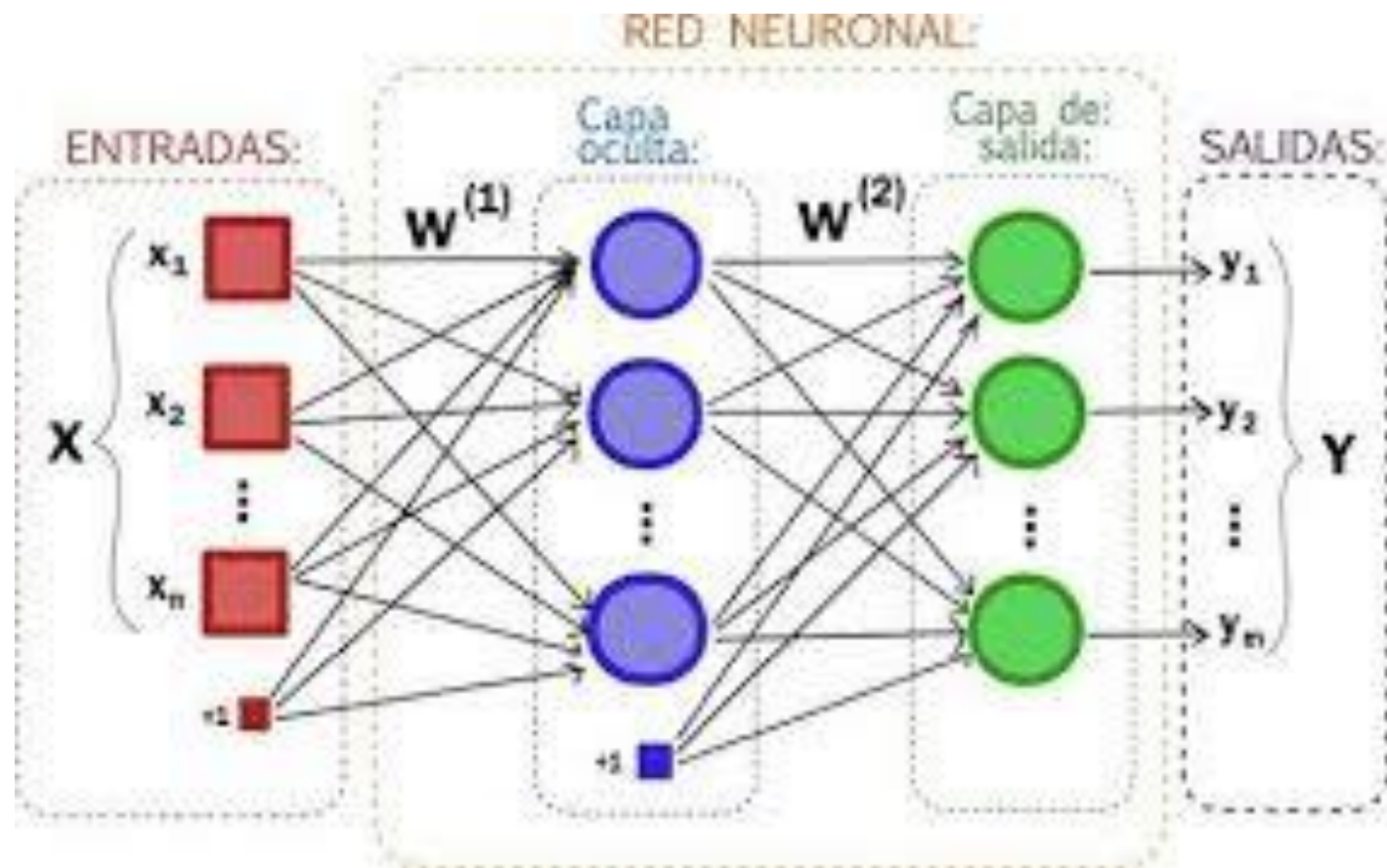


Redes neuronales – Una Neurona



$$f[x1, x2, x3] = Act[p1*x1 + p2*x2 + p3*x3 + p4]$$

Redes neuronales – Perceptrón Multicapa



Redes neuronales – Configuración

QUE COSAS DEBE DECIDIR?

- Cuantas neuronas
- Cuantas capas
- Como se conectan las capas

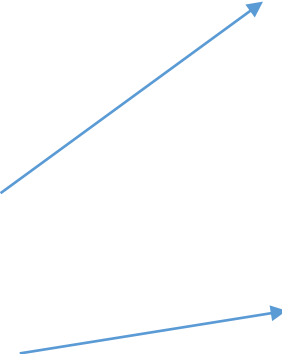
Características estructurales

- Función de activación
- Función de aprendizaje
- Función de error

Características funcionales

Implementación de modelo NN en Python

1. Importar librerías
2. Cargar Datos/selección/renombrar
3. Limpieza de Datos
4. Transformar variables categóricas
5. Partición de conjunto de datos
6. Ajuste del modelo y predicción
7. Cálculo de precisión



```
from sklearn.neural_network import MLPRegressor

#Defino el algoritmo a utilizar
MLP = MLPRegressor(random_state=1, max_iter=500)

#Entreno el modelo
MLP.fit(X_train, y_train)

#Realizo una predicción
Y2_pred= MLP.predict(X_test)

MLP.score(X_test, y_test)
```