



UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH



**Barcelona
Supercomputing
Center**
Centro Nacional de Supercomputación



XNDL - Lab

Regression

Methodology

- ❖ For each problem
 - ❖ Obtain a baseline (linear regression)
 - ❖ Try to beat it with a MLP
-
- ❖ Deliver a 100 words report on the lessons learnt
 - ❖ Include a table with baseline and MLP

Regression problem

❖ Metrics

- `mean_squared_error` (useful for loss)
- `root_mean_absolute_error` / `mean_absolute_error` (interpretable)
- `r2_score` (normalized)

❖ Output activation

Baby steps

- ❖ Data nature & dimensionality
- ❖ Loss function & output
- ❖ Define a baseline
- ❖ Fix a batch size (stochasticity vs efficiency)
- ❖ Start small (as in one neuron)
 - Early stop (update)
 - LR
 - Act. Func. (weight init!)
 - Mom
- ❖ Grow

Datasets

- ❖ Try to beat linear regression on 3 datasets
 - Diabetes (sklearn)
 - California housing (sklearn)
 - Circle.py
- ❖ val split = 20%

Tips

- ❖ relu -> he_normal init
- ❖ tanh /sigmoid -> glorot_init
- ❖ Beware of relu's loss explosions
- ❖ Explore data linearity

Recommended imports

- ❖ `import sys`
- ❖ `from keras.models import Sequential`
- ❖ `from keras.layers import Dense`
- ❖ `from keras.optimizers import SGD, Adam`
- ❖ `import numpy as np`
- ❖ `import matplotlib.pyplot as plt`
- ❖ `from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score`
- ❖ `from sklearn.model_selection import train_test_split`
- ❖ `from keras.callbacks import EarlyStopping`
- ❖ `import tensorflow as tf`
- ❖ `from keras.initializers import he_normal, glorot_normal`
- ❖ `from sklearn.linear_model import LinearRegression`



UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH



**Barcelona
Supercomputing
Center**
Centro Nacional de Supercomputación



The end