Functions

training(*Database, Type, Outputnet, nbtraining, path*)

Allows you to train the neural network for a given parameter.

Settings

- **database**-file that contains experimental data
- **kind**-name of the parameter you wish to train (Young, Hardness, XRD or EBSD)
- **Outputnet**-name of the file where you want to store the training data
- **nbtraining**-number of loops per pack of 5 folds.
- **path**-path to access the folder

Exit

2 files: A .mat file which stores the parameters of the neural network and a .csv file which stores the RMSE for each test.

RMSE_trace(*path, Young, Hardness, XRD, EBSD)*

Allows you to plot all the RMSEs obtained for each of the neural networks in the form of a histogram. This function is useful when a large number of training sessions are carried out.

Settings

- **Path**–path to access the folder
- **Young, Hardness, XRD, EBSD**–name of the neural networks for opening spreadsheets containing RMSE.

Exit

Displays 4 histograms on a single figure

Comparison_exp_IA(*Filename, path*)

Plot for each of the neural networks the predicted values as a function of the experimental values. If the predicted values are correct, the result should be a line with coefficient 1.

Settings

- **Filename**–contains the name under which the predicted values will be saved
- **Path**-path to access the folder

Exit

Shows graphs of predicted values versus experimental values in a figure.

prediction_exp(*Filename, File1, File2, File3, File4, path*)

This function allows you to predict parameter values for compositions used experimentally. This will be useful by the next function “comparison_exp_IA”.

Settings

- **Filename**–contains the name of the file in which the predicted values will be saved
- **File**–name of the .mat file which contains the information of a neural network. 4 files because 4 different neural networks.
- **Path**-path to access the folder

Exit

A .csv file which contains the predicted values for the 4 quantities.

connection prediction(*percentage, path*)

This function creates a matrix_connection{percentage}.csv file in the “matrix_connection” folder containing the number of neighbors of each point and which is used to create the Delaunay representation when calling the function **Representationpredictions***(Filename,percentage,point_size,path).*

It also creates a composition_prediction{percentage}.csv file in the “compo_prediction” folder which contains 5 columns corresponding to the atomic percentages of the 5 elements. The compositions vary depending on the percentage chosen.

Settings

- **percentage**–desired percentage for calculating the predicted values and for the different representations
- **Path**-path to access the folder

Exit

.csv files: matrix_connection{percentage}.csv and composition_prediction{percentage}.csv

prediction(*Filename, percentage, File1, File2, File3, File4, path*)

This function allows you to predict the parameter values for the compositions corresponding to the chosen percentage. Predictions are saved in a .csv file

Settings

- **Filename**–contains the name of the file in which the predicted values will be saved.

- **File 1**– name of the .mat file which contains the information of the neural network used for predicting the hardness modulus.
- **File 2**– name of the .mat file which contains the information of the neural network used for the prediction of the Young module.
- **File 3**– name of the .mat file which contains the information of the neural network used for phase prediction using the XRD method.
- **File 4**– name of the .mat file which contains the information of the neural network used for phase prediction according to the EBSD method.
- **Path**-path to access the folder.

Exit

A .csv file which contains the predicted values for the 4 quantities.

deltaH(*Filename, path*)

This function allows you to calculate the ductility index for each composition from the Galanov model.

Settings

- **Filename**–the name of the file containing the values of the Young's modulus and the hardness modulus from which we wish to calculate the ductility index.
- **Path**-path to access the folder.

Exit

Adds a column to the .csv file created when calling the function **prediction**

Pareto Optimization(*Filename, tolerance, path*)

This function makes it possible to determine the Pareto front on the one hand for the amochromatic phases and on the other hand for the crystalline phases. It creates a final file part _opt.csv which contains only the compositions corresponding to the Pareto front or those close to the Pareto front according to the desired tolerance. The function sorts the points from largest hardness modulus to smallest and determines the maximum ductility index value for each modulus. The compositions corresponding to the Pareto front (optimal) are coded by “1” and the transient compositions are coded by “0”.

Settings

- **FilenameInput**–the name of the file containing the compositions, the hardness modulus and the ductility index from which we wish to determine the Pareto front and the transitional points.
- **tolerance**-tolerance chosen to determine the transient points.
- **Path**-path to access the folder.

Exit

Creates a {FilenameInput}_opt.csv file containing the optimal and transient compositions.

Representationprediction(*Filename, percentage, point_size, path*)

Displays two plots, one corresponding to the Young's modulus and the other to the hardness module as a function of the composition using Delaunay triangulation.

Settings

- **FilenameInput**–the name of the file containing the compositions, the hardness modulus and the ductility index from which we wish to determine the Pareto front and the transitional points.
- **percentage**-percentage that was used to write the file corresponding to FilenameInput
- **point_size**-Size of points you want to use.
- **Path**-path to access the folder.

Exit

Plots of Young's modulus and hardness modulus as a function of composition.

Representationdataexp(*Filename_exp,path*)

This function takes as input the file containing the experimental values of Young's modulus and hardness and plots them against the composition

Settings

- **Filename**–the name of the file containing the compositions, the hardness modulus and the ductility index from which we wish to determine the Pareto front and the transitional points.
- **Path**-path to access the folder.

Exit

Plot of experimental values of Young's modulus and hardness in depending on the composition.

Representationphase(*Filename,path*)

This function takes as input the file containing the compositions and predictions, in particular the phase type predictions and plots the phase type in the Delaunay representation.

Settings

- **Filename**–the name of the file containing the compositions, the hardness modulus and the ductility index from which we wish to determine the Pareto front and the transitional points.
- **Path**-path to access the folder.

Exit

Plot of the type of phase (amorphous or crystalline) depending on the composition.

Representationpareto(*Filename,path*)

This function takes as input the file containing the compositions, predictions and ductility index in order to plot the hardness modulus against the ductility index. To trace the optimal and transient compositions, the function finds the _opt.csv file corresponding to Filename.

Settings

- **Filename**–the name of the file containing the compositions, the hardness modulus and the ductility index from which we wish to determine the Pareto front and the transitional points.
- **Path**-path to access the folder.

Exit

Plots of hardness modulus as a function of ductility index for normal compositions, optimal compositions (Pareto front) and transitional compositions for the amorphous phases on the one hand and crystalline phases on the other.