



# AI Without Borders

## Set I

This problem statement is the exclusive and proprietary property of AI Without Borders, Inc. Any unauthorized use or reproduction of this information without the prior consent of AI Without Borders, Inc. is strictly prohibited. © 2019, AI Without Borders, Inc. All rights reserved.

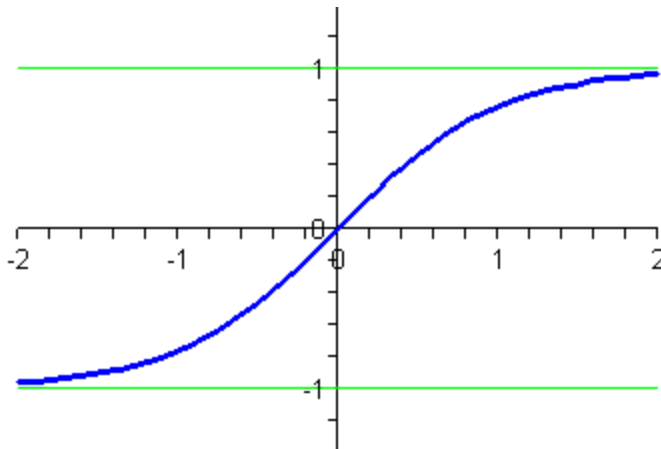
Note: You have three questions and the level of difficulty **increases** per question. Please check the mock notebook provided to get more color on how your deliverable needs to be structured.

All the data you need can be found at this link:

[https://github.com/GaganaB/AI\\_Without\\_Borders/tree/master/Set\\_1/Train](https://github.com/GaganaB/AI_Without_Borders/tree/master/Set_1/Train)

### Question 1:

**Question:** Given a set of numbers, write a formula so that the output values adhere to the structure of the graph below



**Input:** an array of 10000 random numbers

**Output:** an array of 10000 numbers after the formula is applied.

**Hint:** It's that orange juice but instead of the last letter, it's the next letter.

**Expected Code and Output:** Code should take in a file like <input\_file\_name>.txt that contains 10000 numbers, line separated and return an array with your formula applied. Your test\_function (mock in the sample notebook) should take in file like random\_numbers.txt that contains 10000 numbers and return your output array.

## Question 2:

**Question:** Give a set of reviews and their labels (positive/negative), build a model to classify unseen reviews into those classes.

**Input:** A review

**Output:** positive/negative (or) the model's performance on the test set

**Hint:** Cmon, it's all about the words that make you happy vs. words that make you sad.

**Expected Code and Output:** Code should take in a <test\_set>.csv file with one column, *text*, and return the dataframe with two columns, *text*, *predicted\_label*. Your test\_function should take in a dataframe with one column, *text*, and return the same dataframe with your predicted labels under a new column named *predicted\_label*.

## Question 3:

**Question:** Given an image and a set of images, use the former to extract the object to detect the same object in the latter.

**Input:** An image with an object and/or the object.

**Output:** Normalized x and y coordinates of the object location.

**Hint:** Greyscale. Find. Cut. Greyscale. Use Cut. Find. Point. Pick.

**Expected Code and Output:** Code should take in a <test\_image>.jpg file and return the normalized x and y coordinates of the object location. Your test\_function (mock in the sample notebook) should take in an image file image.jpg and return the normalized x and y coordinates of the object location.