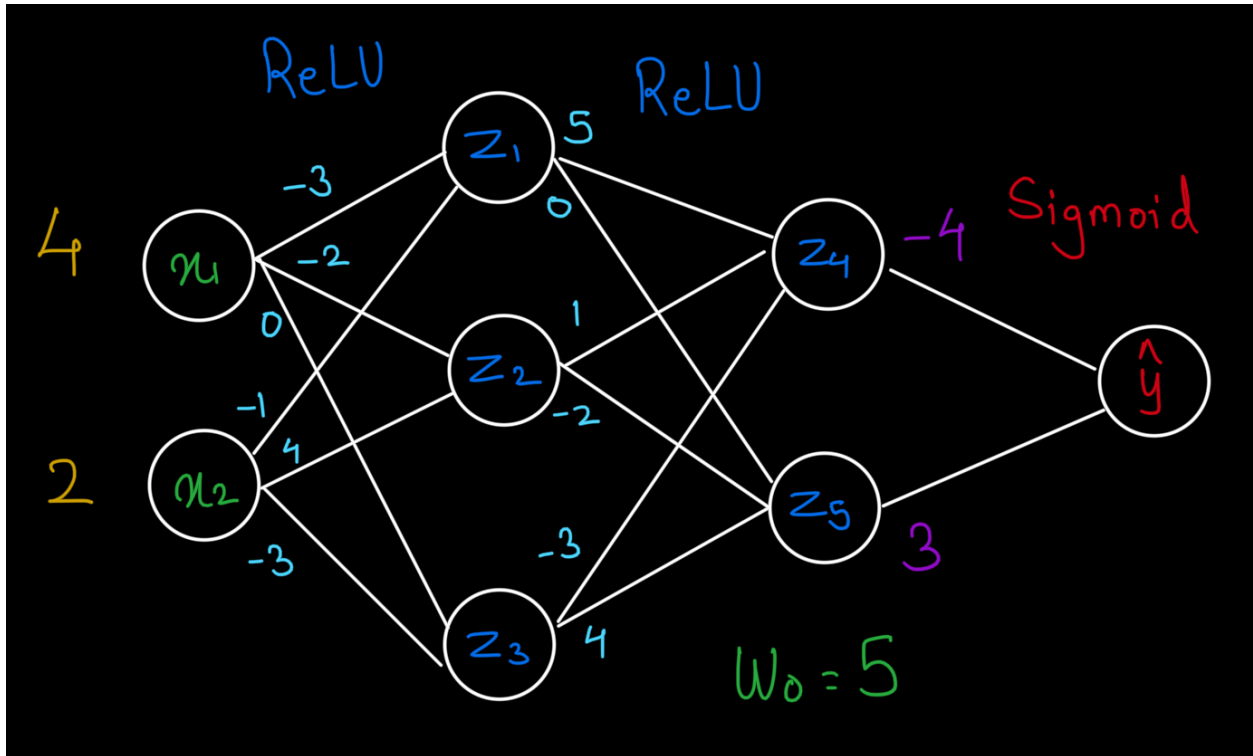# Assignment 1.1

Estimate the value of ybar in the following neural network,

Assignment 1.2

Problem Statement - HR Analytics

Companies get a large number of signups for their training. Now, the company wants to connect these enrollees with their clients who are looking to hire employees working in the same domain. Before that, it is important to know which of these candidates are really looking for a new job. They have student information related to demographics, education, experience and features related to training as well.

To understand the factors that lead a person to look for a job change, the agency wants you to design a deep learning dense neural network model that uses the current credentials/demographics/experience to predict the probability of an enrollee to look for a new job.

Data Link:

Task:

1. Use only the train_jqd04QH.csv file for modeling.
2. Preprocess the data in following ways,
    a. Fill the company_size NA to an "unknown" label, convert all other labels of company_size into "CS_Tier<num>" format such that tier 1 is small size and last tier as large size.
    b. Fill the gender NA to the "unknown" label.
    c. Fill the relevent_experience NA to the "unknown" label, convert all "Has…" as "RE_Yes", and "No…" as "RE_No".
    d. Fill the enrolled_university NA to the "unknown" label, make all other labels in short form, same goes for major_discipline and company_type.
    e. Fill the education_level NA to the "0" label, give the integer rank to the education_level, in which "Primary…" as 1 and "PhD" as highest.
    f. Preprocess experience cautiously as required, create 7 bins for the experience column.
    g. Preprocess last_new_job as per your requirement.
    h. Encode the city label.
    i. Compute the log of training_hours.
3. Perform following EDA,
    a. Visualize and explain how education_level, enrolled_university, and major_discipline are behaving with the target variable using parallel_categories.
    b. Create a crosstab with any input variable and target variables, and explain the result.
4. Split your data into training and testing with 80:20 ratio.
5. Create a dense neural network using Tensorflow with 4 hidden layers, and 1 output neuron with sigmoid activation on a training set.
6. Fit dropout layer with Early stopping, use SGD optimizer and binary cross-entropy.
7. Calculate AUC-ROC score for above DNN on a testing set.
8. List out your key takeaways. Prepare an executive summary within the notebook.