The purpose of this homework is creating a recommendation system using the UCB algorithm that recommends a joke to the user based on his available ratings to some other jokes.

Jester data. The Jester dataset provides continuous ratings in [-10, 10] for 40 jokes from a total of 19181 users. We take the first d = 32 of the ratings as the context of the user, and the remaining k = 8 as the arms. The agent recommends one joke, and obtains the reward corresponding to the rating of the user for the selected joke. Use the following code along with the file which you can download here. It returns three arrays of length 19181:

```
import numpy as np
import tensorflow as tf

def sample_jester_data(file_name, context_dim = 32, num_actions = 8, num_contexts = 19181,
    shuffle_rows=True, shuffle_cols=False):
    """Samples bandit game from (user, joke) dense subset of Jester dataset.
    Args:
        file_name: Route of file containing the modified Jester dataset.
        context_dim: Context dimension (i.e. vector with some ratings from a user).
        num_actions: Number of actions (number of joke ratings to predict).
        num_contexts: Number of contexts to sample.
        shuffle_rows: If True, rows from original dataset are shuffled.
        shuffle_cols: Whether or not context/action jokes are randomly shuffled.
    Returns:
        dataset: Sampled matrix with rows: (context, rating_1, ..., rating_k).
        opt_vals: Vector of deterministic optimal (reward, action) for each context.
  """
    np.random.seed(0)
    with tf.gfile.Open(file_name, 'rb') as f:
        dataset = np.load(f)

    if shuffle_cols:
        dataset = dataset[:, np.random.permutation(dataset.shape[1])]
    if shuffle_rows:
        np.random.shuffle(dataset)
    dataset = dataset[:num_contexts, :]

    assert context_dim + num_actions == dataset.shape[1], 'Wrong data dimensions.'
```

```
    opt_actions = np.argmax(dataset[:, context_dim:], axis=1)
    opt_rewards = np.array([dataset[i, context_dim + a] for i, a in enumerate(opt_actions)])
    return dataset, opt_rewards, opt_actions
```

The first array is the rating of all 19181 to all 40 jokes. The second array is the optimal reward corresponding to the optimal action (suggested joke). The third array is the optimal action. Note that we are choosing the optimal joke among the last 8 jokes. Your job is using the first 32 columns of the first array as the context of the user and the remaining 8 columns as actions and their corresponding rewards and:

- Apply the UCB algorithm to the first 18000 rows of the data (You need to tune for the parameter of UCB algorithm).
- Evaluate the trained model on the remaining rows of the data and plot the regret for them.