Exp3 Algorithm:

The parameter gamma is taken as a random quantity between 0 and 1. To deal with Stochastic and Adversarial settings both, pseudo-regret and actual regret are taken into consideration.
First equal weights are assigned to each arm and an arm is drawn at random with the given weights. Then according to the reward observed, the weights are updated for the next iteration. Both the pseudo regret and actual regret are calculated as the difference between the observed reward and the best possible reward.

The process is repeated and cumulative regret across different time horizons is stored in a list to see how the relation between the two.

Explanation of creation of bandit instance:

The bandit instance is created in two ways.

1. **Stochastic setting:** In this the mean reward for each arm is taken at random and the rewards for each arm are generated using a normal distribution with the chosen mean and variance 1.
2. **Adversarial setting:** In this, the reward for each bandit is generated using a uniform distribution (0, i/K), where i= bandit number and K= total number of bandits. This is just done so that the bandits differ by some good margin.

Stochastic Instances:
We need Mean reward values for all the arms and the Reward Table for K arms and T rounds. The mean values are the random values generated and stored in an array. Now, the reward table is generated following a normal distribution with the mean value taken from Mean Table and variance equal to 1.
We used ‘np.random.seed(69)’ to get the same result every time the code is run.

Adversarial Instances:
We now need only the Cost Table to be generated randomly following uniform distribution depending on the number of times a given arm was pulled. The Mean Table now will be based on the outcomes or the costs that we incurred over our previous pulling of the arms. The mean table gets generated from the Cost Table. Thus, this provides more generality to our instances.
We used ‘np.random.seed(69)’ to get the same result every time the code is run.

Reference used:

http://rob.schapire.net/papers/AuerCeFrSc01.pdf

Contribution of each team member:

Salil: Created the bandit instances & Implemented KL UCB algorithm
Priyank: Implemented EXP3 algorithm
Shumail: Implemented KL UCB algorithm