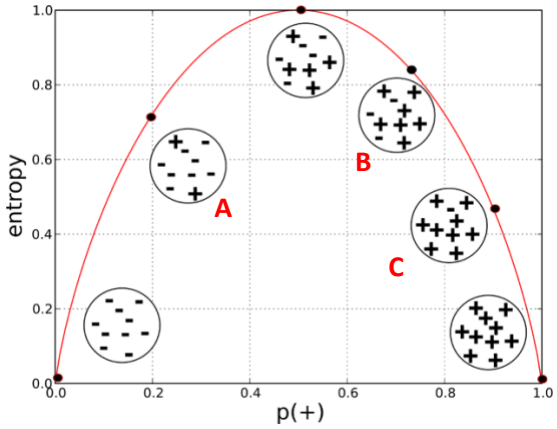
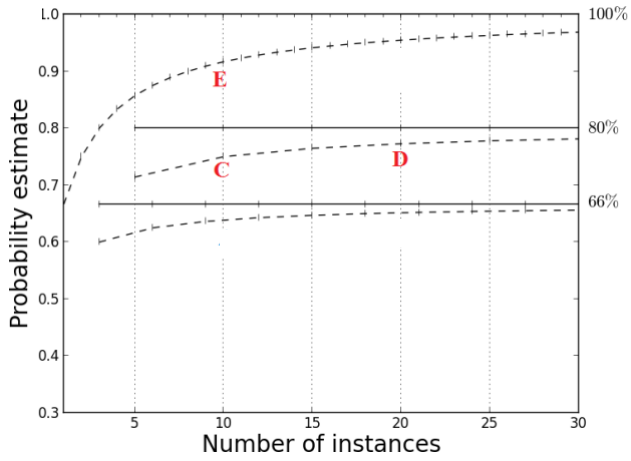
# Assignment 1

***Note**: for questions that require answers with Python, display both the codes and results clearly. You may explain your answers with comments, markdowns or print() function.*

**Question 1**. **[15 points]** The following figure shows the entropy ($y$ axis) for groups with different class distribution ($x$ axis). Please calculate the entropy for group A, B and C respectively with Python, round the results to 2 decimal places.




**Question 2.** **[10 points]** The following figure shows the relationship between total number of instances in a group ($x$ axis) and the estimated positive probability ($y$ axis) based on Laplace-corrected (dashed line) and Frequency-based (solid line) approach. Point C, D and E represent three groups with different total number of instances, and their frequency-based positive probabilities are 80% (for group C & D) and 100% (for group E) respectively. Please calculate the Laplace corrected positive probability for group C, D and E with Python, round the results to 2 decimal places.



# Assignment 1

**Question 3**. **[40 points]** Explore the dataset *churn.csv* and answer the following questions with Python. The target variable is ***LEAVE***.

(a) **[5 points]** Load the data, calculate the entropy for the entire dataset.

(b) **[10 points]** Segment the dataset with attribute ***REPORTED_SATISFACTION***. Create a cross table to calculate the entropy for each child node, save the results in a new column named ***child_entropy***. Then weigh each child node's entropy by the proportion of instances in it and save the results in a column named ***weighted_entropy***. Display the data frame.

(c) **[5 points]** Calculate the information gain for ***REPORTED_SATISFACTION***. Round the result to 4 decimal places.

(d) **[5 points]** Next, create a subset that only contains very unsatisfied customers (i.e., ***REPORTED_SATISFACTION*** == 'very_unsat'), how many customers are there? What is the entropy of this subset?

(e) **[5 points]** Segment the subset created in step (d) into two new child notes according to whether the customer's ***INCOME*** value is above the average or not. How many instances are there in each child node?

(f) **[10 points]** Create a cross table to calculate the information gain for ***INCOME*** on the subset. Display the cross table and the information gain. Round the result to 4 decimal places.

**Question 4**. **[30 points]** We'd like to train a decision tree model to predict whether a customer will leave or not. Continue to work with the dataset *churn.csv* and answer the following questions with Python.

(a) Load the data, remove three string features (i.e., ***REPORTED_SATISFACTION, REPORTED_USAGE_LEVEL, CONSIDERING_CHANGE_OF_PLAN***) and convert the variable ***COLLEAGE*** as numbers. Display the shape of the cleaned dataset.

(b) Prepare the cleaned data by getting features and target variable properly. Split the data into train and test set, with 20% of the data used for model evaluation. Set random seed as 42. Display the shape of all returned datasets.

**Assignment 1**

(c) Train a decision tree classifier on the train set, set *min_samples_leaf* = 1000, *max_depth* = 3, *random_state* = 1. Name the model as ***model2*** and visualize it. (*Hint: you may adjust the figure size to have the tree displayed properly.*)

(d) Apply ***model2*** to predict the class labels and estimate the class probabilities for test set only. Display the results for the first five test instances only.

(e) Check ***model2***'s performance on the test set. Round the result to 2 decimal places.