1 Graph balancing

We are given an unoriented graph with non-negative weights on the edges. Given an orientation, we define the weight of a vertex as the sum of weights of the incoming edges. The goal is to orient all the edges to minimize the weight of the heaviest vertex.

- Formulate an ILP and round the relaxation to get a 2-approximation algorithm.

2 Maximal directed cut

We are given an oriented graph with non-negative weights on the edges. The goal is to find $S \subseteq V$ such that the weight of edges from S to $V \setminus S$ is minimized.

- Formulate a (trivial) $^1/_4$-approximation algorithm.

2.1 Bonus

- Formulate an ILP for this problem, relax it and add a vertex v to S with probability $^1/_4 + {}^{x_v}/_2$. Did you get a $^1/_2$-approximation?

3 Integrality gap

Let OPT and OPT_{LP} be the optimal values (of an instance) of an integer LP and its relaxation respectively. The **integrality gap** of this linear program (instance) is $\frac{\text{OPT}}{\text{OPT}_{LP}}$.

- Show that MAX-SAT has integrality gap at least $^4/_3$.
- Show that vertex cover has a tight integrality gap $2 - {}^1/_n$.

4 Coin flip independence

We flip a fair coin n times. For $i, j \in [n]$, let X_{ij} indicate the event "i-th and j-th coin flips are the same".

- Show that X_{ij} are 2-independent but not fully independent.

5 Uniform independence

Let $X_1, ..., X_k$ be independent from the distribution $\mathcal{U}(1, 4)$, meaning $\forall i \in [k]\ \forall j \in [4] :\ P[X_i = j] = {}^1/_4$.

- Construct as many discrete 2-independent random variables with distribution $\mathcal{U}(1, 4)$ as possible.