

Data-driven Decision Making

Graph problems: Kidney Exchange Programs

João Pedro Pedroso

Kseniia Klimetnova

2025/2026

Last class

- matching problems
- kidney exchange problems

Today's class:

- more on kidney exchange

KEP: cycle formulation

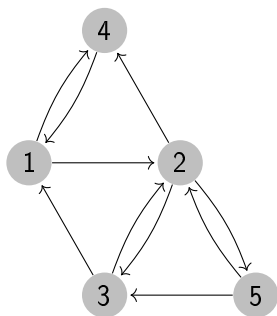
Kidney Exchange Model

- Given:
 - a pool of n incompatible donor-patient pairs
 - the compatibility between all donors and all patients
- find the **maximum number** of kidney exchanges with cycles of size at most k

- Is this problem **easy to solve**?
 - YES, if $k = 2$ or no limit is imposed on the size of the cycles
 - NO, if $k = 3, 4, 5, \dots$
- If $k = 2$ the problem reduces to finding a **maximum matching** in a undirected graph, which can be solved efficiently (Edmonds 1965)
- If no limit is imposed on the size of the cycles the problem can be formulated as an **assignment problem** (can be solved efficiently by hungarian algorithm)
- The problem is NP-hard for $k = 3, 4, 5, \dots$ (hence, no polynomial algorithms are known to solve it)

Mathematical programming formulations

- There are several possibilities for modeling the problem in mathematical programming
- One of the most successful is the **cycle formulation**:
 - enumerate all cycles in the graph with length at most K
 - for each cycle c , let variable x_c be 1 if c is chosen, 0 otherwise
 - every feasible solution corresponds to a set of vertex-disjoint cycles



Cycles for $K = 3$:

- 1 [1,2,3]
- 2 [1,2,4]
- 3 [1,4]
- 4 [2,3]
- 5 [2,5]
- 6 [2,5,3]
- 7 ~~[1,2,5,3]~~

$$\text{maximize } \sum_c w_c x_c \quad (1)$$

$$\text{subject to } \sum_{c:i \in c} x_c \leq 1 \quad \forall i \in V \quad (2)$$
$$x_c \in \{0, 1\} \quad \forall c$$

- case of 0 – 1 weights: $w_c = |c|$, (length of cycle c)
- objective: maximize the weight of the exchange
- constraints: every vertex is at most in one cycle
 - (*i.e.*, donate/receive at most one kidney)
- difficulty: exponential number of variables
 - in our experience: not an issue in practice

- Ethically, a sensitive subject
- Most common: **maximize total number of transplant**
 - easy to explain/justify
 - difficult to dispute
- Our claim: **this is not adequate**
 - KEPs are typically run periodically
 - most of the pairs will eventually be matched, after waiting a few periods
 - hence, we should look for alternatives
 - somehow access the **quality** of the transplants

KEP: possible objectives

Objective: our proposal

- For each pair i
 - for each possible matching j for that pair
 - i 's patient may receive kidney of j 's donor
 - determine **expected survival time** for patient s_{ji}
- Then, maximize $\sum s_{ji}$, for all arcs in the solution
- This can be done by assigning the weight w_c for each cycle:

$$w_c = \sum_{ji \in c} s_{ji}$$

- **Problem:**
 - n^2 weights to determine \rightarrow *not practical, if done manually*
 - solution?

Predicting survival time

- There is a considerable amount of **historical data**
- Idea: **use *historical data* to train a machine learning model**
- This has been done in the past, but to our knowledge not to parameterize a KEP optimization model
- See, eg., Living Kidney Donor Risk Index (LKDPI)
<http://www.transplantmodels.com/lkdpi/>

LKDPI Score:

9

This model calculates a risk score for a recipient of a potential live donor kidney.

Live Donor Characteristics:

Donor age:

43



Donor sex:

male



Recipient sex:

female



Donor eGFR:

95



Donor SBP:

130



Donor BMI:

24



Donor is African-American:

No



Donor history of cigarette use:

No



Donor and recipient biologically related:

Yes



Donor and recipient are ABO incompatible:

No



Donor Weight:

70 kg/155 lb



Recipient Weight:

80 kg/178 lb



Donor and recipient HLA-B mismatches:

1



Donor and recipient HLA-DR

Proposed flow:

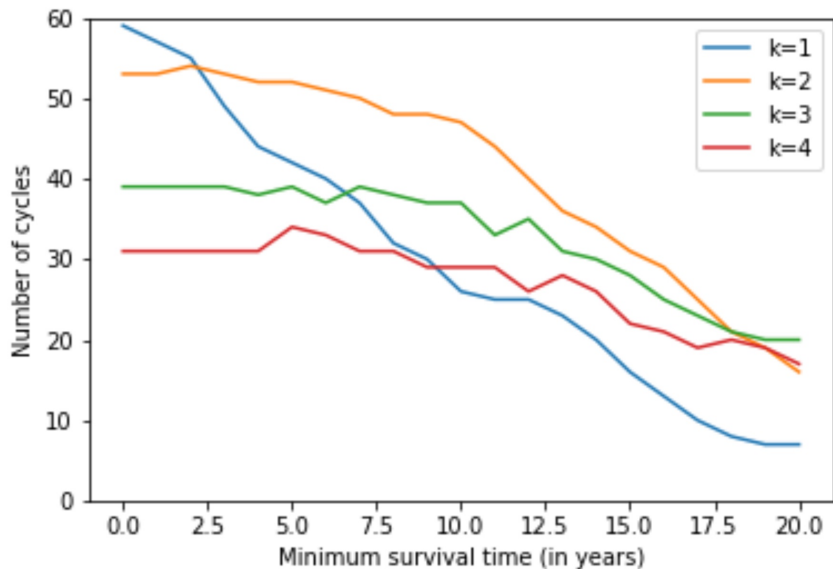
- 1 Use past data to train regression model
 - features: → patients' data
 - output: → observed **survival times**
- 2 Gather information concerning patients in current pool
- 3 Parameterize KEP optimization model using *pool information + predicted survival times*

- **Data:** generated artificially, loosely based on LKDPI
 - *past data* → used for training regression model
 - features + survival time
 - *current pool data* (100 patient/donor pairs)
 - patients/donors features
 - used in regression model
 - predict survival time
 - "true" survival time → "*ground truth*"
used for assessing performance/testing

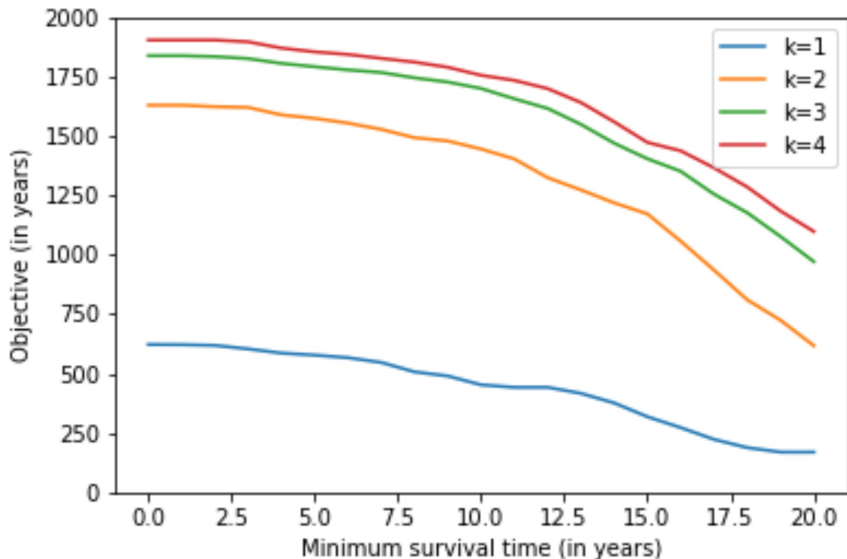
- **Optimization model:**
 - cycle model
 - each cycle's weight w_c : sum of patients' survival times
 - obtained by regression, using donor's and patient's *features*
 - objective: maximize survival time summed for all selected cycles
 - compatible pairs allowed
 - *rationality constraint*:
compatible pairs only accept donors better than their own
 - i.e., leading to longer survival times

- **Assessment:**
 - compare "ground truth" survival times on:
 - solution with ML + KEP vs. "ground truth" + KEP
 - solution with ML + KEP vs. max-transplants KEP
- **Parameter/Ethics:**
 - we must decide the **minimum patient's survival time** allowed in the program. . .

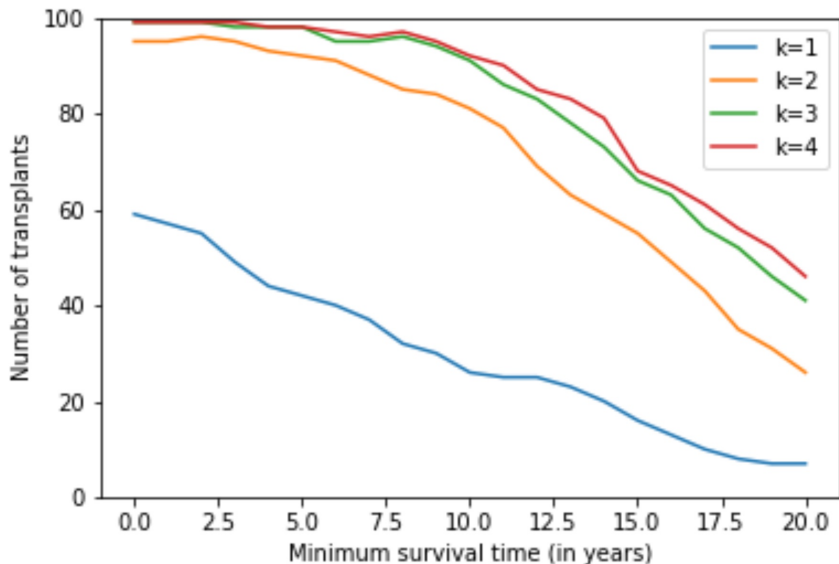
Number of cycles vs. minimum patient's survival time

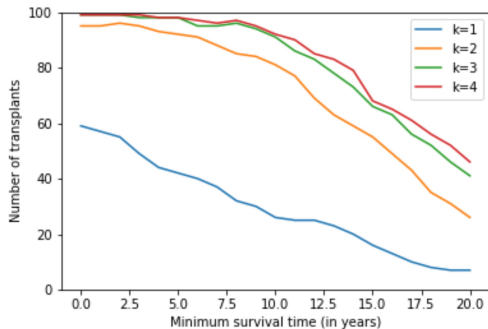
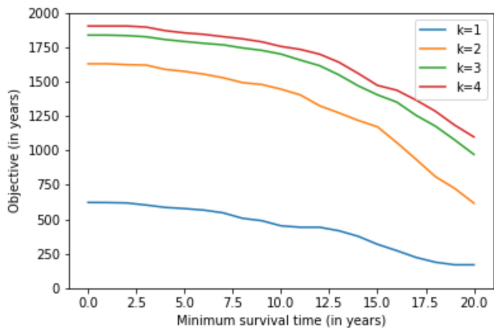


Results: total survival time vs. minimum survival time



Results: number of transplants vs. minimum survival time





Proposed flow: summary

- 1 Use past data to train regression model → **survival times**
- 2 Gather information concerning patients
→ predict **survival time for each possible donor-patient** assignment
- 3 Find all cycles of desired length
→ **filter cycles with unacceptably low-survival** patients
- 4 Prepare KEP optimization model using this information
- 5 Execute optimization model, retrieve and analyse solution
- 6 In the long run: follow transplanted patients, use survival times to improve regression model

Conclusions

- Are current kidney exchange programs wrong?
 - is the objective correct?
- There seems to be room for improvement
 - machine learning + optimization
 - total survival time seems to be a better objective
- Ethics:
 - patient's *minimum acceptable survivable time* should be discussed

This lesson

- matching problems
- kidney exchange problems

Next lessons:

- kidney exchange problems
- machine learning → data-driven components