Sudoku Assignment
SA405, Fall 2018
Instructor: Foraker
Note: Assignment adapted from Sudoku, by Matt Carlyle, PhD

Figure 1: A (very difficult) Sudoku puzzle of order three.

1 Introduction

A *Sudoku puzzle of order n* consists of an $n^2 \times n^2$ table of squares, each of which is either empty or contains one of the integers one through n^2. The goal of a Sudoku puzzle is to fill in the remaining squares with a number between one and n^2 so that each row, each column, and each major $n \times n$ block contains a *permutation* of the numbers $\{1, \ldots, n^2\}$. Figure 1 shows a puzzle of order three; to solve it we must complete each row, column, and major 3x3 block (with bold borders) with a permutation of the numbers $\{1, \ldots, 9\}$.

The assumption we have to make when we try to *solve* a Sudoku puzzle is that there is a unique way to fill in each empty square. If we are tyring to *design* a Sudoku puzzle, we could start with a completely full board and remove values from squares until we get to a point at which removing one more value would yield multiple solutions. To support us in both of these efforts, we present a formulation for puzzles of order three.

Sets {members}

i	Rows, columns (alias j), and values (alias k) $\{1, ..., 9\}$
p	Row and column (alias q) blocks $\{1, 2, 3\}$

Data [units]

$\overline{x}_{ij}$	Given value in row i, column j [integer 0-9]

Decision Variables [units]

$Y_{i,j,k}$	=1 if row i column j has value k, 0 otherwise [binary]

Formulation

$$
\begin{aligned}
\max_{Y} \quad & \sum_{i,j,k} Y_{i,j,k} & & (1)\\
\text{s.t.} \quad & \sum_{j} Y_{i,j,k} = 1 & \forall i,k & \quad (2)\\
& \sum_{i} Y_{i,j,k} = 1 & \forall j,k & \quad (3)\\
& \sum_{\substack{3(p-1)<i\leq 3p \\ 3(q-1)<j\leq 3q}} Y_{i,j,k} = 1 & \forall p,q,k & \quad (4)\\
& \sum_{k} Y_{i,j,k} = 1 & \forall i,j & \quad (5)\\
& Y_{i,j,k} = 1 & \forall\ \overline{x}_{ij} = k > 0 & \quad (6)\\
& Y_{i,j,k} \text{ binary} & \forall i,j,k &
\end{aligned}
$$

Discussion

Objective (1) is aribitrary, since we are assuming the solution is unique. Constraints (2) ensure that each row i uses value k exactly once, while Constraints (3) ensure the same for each column j. Constraints (4) ensure that each major block uses value k exactly once. Constraints (5) guarantee that each square has exactly one value, and Constraints (6) fix the squares for which values were already provided.

2 Homework

Submit your model file, data file, and Excel file via Blackboard by 2200 on 26 October 2017.

1. Implement this formulation in GUSEK, and use GLPK to solve the puzzle in Figure 1.

2. Implement an interface in Excel/VBA: give a 9x9 grid, and, with one button push, create a GUSEK data file and run GUSEK, and then with a second button push read in the solution.

   Parts 3 and 4 below are OPTIONAL. To get the BONUS: Submit your modified model file, data file, and Excel file via Blackboard by 2200 on 26 October 2017.

3. Modify the formulation and GUSEK to determine whether the solution you find is unique. (Hint: modify your objective function to try to *change* as many cells as possible from the previous solution. This will require some extra parameters.)

4. How many filled-in squares can you remove from Figure 1 and keep a unique solution? This can be done with a macro in VBA (or "by hand"), using parts 2 and 3 above.