Read: Chapter 3.1–3.4 of Robeva/Hodge: *Inferring the topology of gene regulatory networks: an algebraic approach to reverse engineering.* By B. Stigler and E. Dimitrova, pages 75–90.

Exercises.

1. Consider the following time-series over $\mathbb{F}_2$:

$$(1,1,1) \xrightarrow{f} (1,1,0) \xrightarrow{f} (0,0,1) \xrightarrow{f} (0,0,1)\,.$$

   (a) How many Boolean networks $f = (f_1, f_2, f_3)$ fit the following data? By inspection, find two of them. Express your answer using Boolean logic and as polynomials in $\mathbb{F}_2[x_1, x_2, x_3]/\langle x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3\rangle$.

   (b) Write down the *vanishing ideal*, I. That is, the set of all triples of polynomials $f = (f_1, f_2, f_3)$ that vanish on the input data. (Giving a generating set for I is fine.)

   (c) Write down the *model space* $F_1 \times F_2 \times F_3$ consisting of all functions that fit the data. [That is, write a formula for F_j for each $j = 1, 2, 3$.]

2. Consider the following *time series* in a 3-node polynomial dynamical system over $\mathbb{F}_3$:

$$(1,1,1) \xrightarrow{f} (2,0,1) \xrightarrow{f} (2,0,0) \xrightarrow{f} (0,2,2) \xrightarrow{f} (0,2,2)\,.$$

   For reference, here are the input vectors $\mathbf{s}_i$ and output vectors $\mathbf{t}_i$:

$$\begin{aligned}
\mathbf{s}_1 &= (s_{11}, s_{12}, s_{13}) = (1,1,1)\,, & \mathbf{t}_1 &= (t_{11}, t_{12}, t_{13}) = (2,0,1)\,,\\
\mathbf{s}_2 &= (s_{21}, s_{22}, s_{23}) = (2,0,1)\,, & \mathbf{t}_2 &= (t_{21}, t_{22}, t_{23}) = (2,0,0)\,,\\
\mathbf{s}_3 &= (s_{31}, s_{32}, s_{33}) = (2,0,0)\,, & \mathbf{t}_3 &= (t_{31}, t_{32}, t_{33}) = (0,2,2)\,,\\
\mathbf{s}_4 &= (s_{41}, s_{42}, s_{43}) = (0,2,2)\,, & \mathbf{t}_4 &= (t_{41}, t_{42}, t_{43}) = (0,2,2)\,.
\end{aligned}$$

   (a) Find polynomials f_1, f_2, f_3 in $\mathbb{F}_3[x_1, x_2, x_3]$ that fit the data. That is, $f_j(\mathbf{s}_i) = t_{ij}$ for all $i = 1, 2, 3, 4$.

   (b) For each $i = 1, 2, 3, 4$, write down the ideal $I_i = I(\mathbf{s}_i)$ of polynomials that vanish on the data point $\mathbf{s}_i$.

   (c) Use the following commands in Macaulay2 to compute the ideal I of polynomials that vanish on *all* of the input data points.

```
R = ZZ/3[x1,x2,x3,MonomialOrder=>Lex];
I = intersect{I1, I2, I3, I4};
```

   Compute a Gröbner basis $\mathcal{G}$ of I.

   (d) Write the *model space* of the time series using your answer to Part (a) as the particular solution.

   (e) Compute the normal form of f_1, f_2, f_3 with respect to $\mathcal{G}$ by reducing them modulo the ideal I. Write the model space using this particular solution.

   (f) Repeat Parts (c)–(e) using graded lex (`GLex`), graded reverse lex (`GRevLex`), and reverse lex (`RevLex`).

Summary of relevant literature.

The reverse engineering algorithm was originally published in [LS04]. As a pre-processing step, sometimes (continuous) data needs to be discritized. One algorithm to do this was published in [DLML10]. In [AFD$^+$06], the authors applied algebraic reverse engineering techniques to time-series data from protein signal transduction networks. Specifically, they used this to identify protein dependencies.

Other (non-algebraic) algorithms to reverse engineer gene networks include [BdlFM02], [GF05], [BBAIDB07], [THSH05].

References

[AFD$^+$06] E.E. Allen, J.S. Fetrow, L.W. Daniel, S.J. Thomas, and D.J John. Algebraic dependency models of protein signal transduction networks from time-series data. *J. Theor. Biol.*, 238(2):317–330, 2006.

[BBAIDB07] M. Bansal, V. Belcastro, A. Ambesi-Impiombato, and D. Di Bernardo. How to infer gene networks from expression profiles. *Mol. Syst. Biol.*, 3(1):78, 2007.

[BdlFM02] P. Brazhnik, A. de la Fuente, and P. Mendes. Gene networks: how to put the function in genomics. *TRENDS Biotech.*, 20(11):467–472, 2002.

[DLML10] E.S. Dimitrova, M.P.V. Licona, J. McGee, and R. Laubenbacher. Discretization of time series data. *J. Comp. Biol.*, 17(6):853–868, 2010.

[GF05] T.S. Gardner and J.J. Faith. Reverse-engineering transcription control networks. *Phys. Life Rev.*, 2(1):65–88, 2005.

[LS04] R. Laubenbacher and B. Stigler. A computational algebra approach to the reverse engineering of gene regulatory networks. *J. Theor. Biol.*, 229(4):523–537, 2004.

[THSH05] V. Thorsson, M. Hornquist, A.F. Siegel, and L. Hood. Reverse engineering galactose regulation in yeast through model selection. *Stat. Appl. Genet. Mol. Biol*, 4:28, 2005.