CS 6550: Randomized Algorithms March 1, 2019

Problem Set 3

Instructor: Eric Vigoda *STUDENT NAME*

Problem 1 Pairwise Independence

Let $X_1, \ldots, X_n$ be pairwise independent random variables, and $X = \sum_{i=1}^n X_i$. Show that

$$\mathrm{Var}(X) = \sum_{i=1}^n \mathrm{Var}(X_i).$$

Well, recall that

$$\begin{aligned}
\mathrm{Var}(X) &= E[X^2] - E[X]^2 \\
&= E\Big[\big(\sum_{i=1}^n X_i\big)^2\Big] - \Big(E\Big[\sum_{i=1}^n X_i\Big]\Big)^2 \\
&= E\Big[\sum_{i=1}^n \big(\sum_{j=1}^n X_i X_j\big)\Big] - \Big(E\Big[\sum_{i=1}^n X_i\Big]\Big)^2 \\
&= \sum_{i=1}^n \big(\sum_{j=1}^n E[X_i X_j]\big) - \Big(\sum_{i=1}^n E[X_i]\Big)^2 \\
&= \sum_{i=1}^n \big(\sum_{j\neq i} E[X_i]E[X_j]\big) + \sum_{j=1}^n E[X_j^2] - \sum_{i=1}^n \big(\sum_{j=1}^n E[X_i]E[X_j]\big) \\
&= \sum_{i=1}^n \big(\sum_{j\neq i} E[X_i]E[X_j]\big) + \sum_{j=1}^n E[X_j^2] - \sum_{i=1}^n \big(\sum_{j\neq i} E[X_i]E[X_j]\big) - \sum_{j=1}^n \big(E[X_j]\big)^2 \\
&= \sum_{j=1}^n E[X_j^2] - E[X_j]^2 \\
&= \sum_{j=1}^n \mathrm{Var}(X_j)
\end{aligned}$$

Problem 2 Random Subset

(a) Let S, T be two disjoint subsets of a universe U such that $|S| = |T| = n$. Suppose we select a random subset $R \subseteq U$ by independently sampling each element of U with probability p; that means, for each element i of U independently we include i in R with probability p. We say that the random subset R is good if the following two conditions hold: $R \cap S = \emptyset$ and $R \cap T = \emptyset$. Show that for $p = 1/n$, the probability that R is good is larger than some positive constant.

Say $S = \{s_1, s_2, \ldots, s_n\}$ and $T = \{t_1, t_2, \ldots, t_n\}$, also noting that $s_i \neq t_j$ for all $i, j \in [n]$. The probability that R is good is

$$\begin{aligned} P(R \cap S = \emptyset \text{ and } R \cap T = \emptyset) &= P(s_i \notin R \forall i \in [n] \text{ and } t_i \notin R \forall i \in [n]) \\ &= (1-p)^{2n} = (1 - \frac{1}{n})^{2n} \end{aligned}$$

We wish to produce $c \in \mathbb{R}_+$ such that $(1 - \frac{1}{n})^{2n} \geq c$ for all $n \in \mathbb{N}_{\geq 2}$. We first compute the derivative of $f(x) = (1 - \frac{1}{x})^{2x}$. We get that

$$log_e(f(x)) = (2x)(log_e(1 - \frac{1}{x}))$$

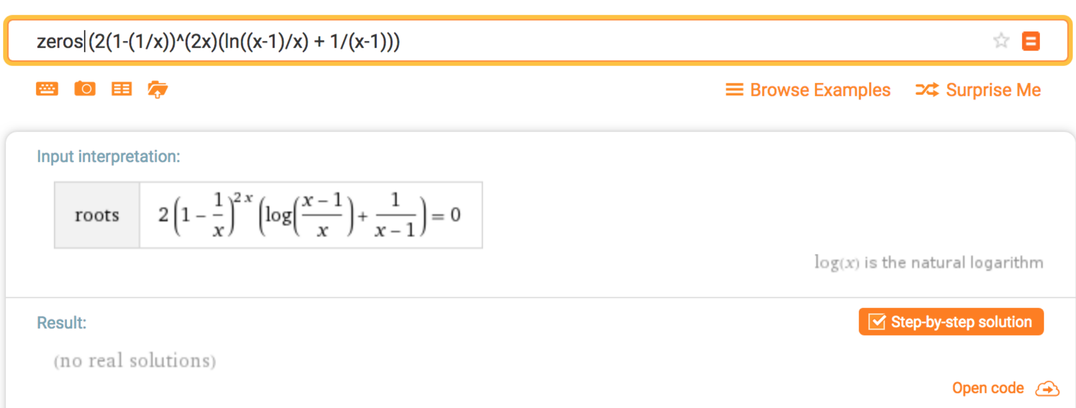
so that

$$\begin{aligned} \frac{d}{dx}(log_e(f(x))) &= \frac{d}{dx}((2x)(log_e(1 - \frac{1}{x}))) \\ \frac{1}{f(x)}(f'(x)) &= 2(log_e(1 - \frac{1}{x})) + (2x)\frac{1}{1 - \frac{1}{x}}(\frac{1}{x^2}) \end{aligned}$$

which gives that

$$\begin{aligned} f'(x) &= \Big(f(x)\Big)\Big(2(log_e(1 - \frac{1}{x})) + (2x)\frac{1}{1 - \frac{1}{x}}(\frac{1}{x^2})\Big) \\ &= \Big(2f(x)\Big)\Big(log_e(\frac{x-1}{x}) + \frac{1}{x-1}\Big) \\ &= \Big(2f(x)\Big)\Big(log_e(\frac{x-1}{x}) + \frac{1}{x-1}\Big) \\ &= \Big(2(1 - \frac{1}{x})^{2x}\Big)\Big(log_e(\frac{x-1}{x}) + \frac{1}{x-1}\Big) \end{aligned}$$

Next, one notes that the derivative $f'(x)$ has no zeros on the interval $[2, \infty)$.



Then, since $f'(2) \geq 0$ (shown below)

Input interpretation:

$$2\left(1-\frac{1}{x}\right)^{2x}\left(\log\left(\frac{x-1}{x}\right)+\frac{1}{x-1}\right) \text{ where } x = 2$$

Open code

$\log(x)$ is the natural logarithm

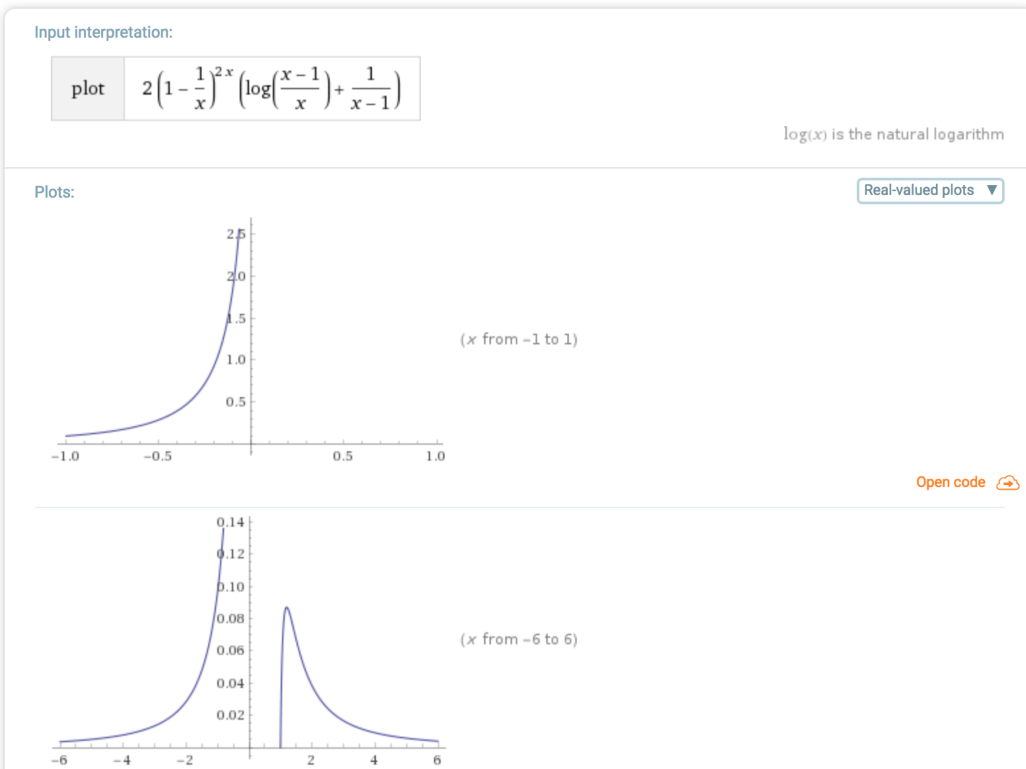
Result:

$$\frac{1}{8}(1-\log(2))$$

Alternate form:

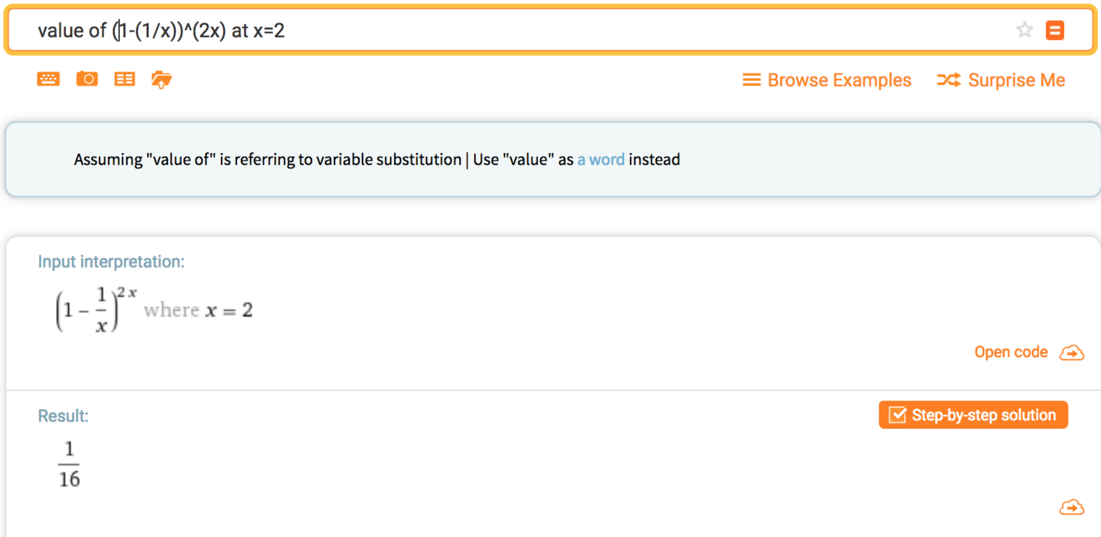
$$\frac{1}{8}-\frac{\log(2)}{8}$$

and f' continuous on $[2, \infty)$,



this means that the derivative f' is always positive on the interval $[2, \infty)$ (since as stated before f' has no zeros on this interval and therefore cannot change sign). So, the function f is strictly increasing on $[2, \infty)$, which implies that $f(x) > f(2)$ for all $x > 2$.

One finally calculates $f(2) = \frac{1}{16}$.



Then, the probability that R is good is always greater than $\frac{1}{17}$.

(b) Suppose now that the random subset R is chosen by sampling the elements of U with only pairwise independence; that means, for each element i of U let $X_i = 1$ if $i \in R$ and $X_i = 0$ otherwise (so, $\Pr(X_i = 1) = p$ and $\Pr(X_i = 0) = 1 - p$), and $\{X_i\}_{i \in U}$ are only pairwise independent instead of mutually independent as in (a). Show that for a suitable choice of the value of p, the probability that R is good is larger than some positive constant.

$Pr(R \text{ is good}) = Pr(R \cap S \neq \emptyset \text{ and } R \cap T \neq \emptyset) = Pr(X_i = 0 \text{ for all } i \in S \cup T) = Pr(\sum_{i \in S \cup T} X_i = 0) = 1 - Pr(\sum_{i \in S \cup T} X_i \geq 1)$. Now, one uses Markov's inequality.

$$\begin{aligned} Pr(\sum_{i \in S \cup T} X_i \geq 1) &\leq \frac{E[\sum_{i \in S \cup T} X_i]}{1} \\ &= \sum_{i \in S \cup T} E[X_i] \\ &= \sum_{i \in S \cup T} P(X_i = 1) \end{aligned}$$

Now, using a constant probability $p(n)$ such that $P(X_i = 1) = p(n)$ for all i gives

$$\begin{aligned} Pr(\sum_{i \in S \cup T} X_i \geq 1) &\leq \sum_{i \in S \cup T} P(X_i = 1) \\ &= \sum_{i \in S \cup T} p(n) \\ &= 2np(n) \\ Pr(\sum_{i \in S \cup T} X_i = 0) &= 1 - Pr(\sum_{i \in S \cup T} X_i \geq 1) \\ &\geq 1 - 2np(n). \end{aligned}$$

Then, setting $p(n) = \frac{1}{5n}$ gives

$$\begin{aligned} Pr(\sum_{i \in S \cup T} X_i = 0) &= 1 - Pr(\sum_{i \in S \cup T} X_i \geq 1) \\ &\geq 1 - \frac{2}{5} = \frac{3}{5} > \frac{1}{2}. \end{aligned}$$

So, using $p(n) = \frac{1}{5n}$ gives that R is good with probability $> \frac{1}{2}$ for any $n \in \mathbb{N}$.

Problem 3 MAX-SAT

Recall that in the MAX-SAT problem, we are given a set of m clauses $C_1, \dots, C_m$ in conjunctive normal form over a set of n variables $x_1, \dots, x_n$, and our goal is to find a truth assignment for the n variables that satisfies as many of the clauses as possible. We assume each clause has at least one term in it, and all the variables in a single clause are distinct. If we have a clause that consists only of a single term (e.g., a clause consisting just of x_1, or just of $\overline{x_2}$), then there is only a single way to satisfy it. If we have two clauses such that one consists of just the term x_i, and the other consists of just the negated term $\overline{x_i}$, then this is a pretty direct contradiction.

Assume that our instance has no such pair of "conflicting clauses"; that is, for no variable x_i do we have both a clause $C = \{x_i\}$ and a clause $C' = \{\overline{x_i}\}$. Prove that there exists a truth assignment for the variables that satisfies at least $0.6m$ clauses.

First, separate the set of clauses into C, the set of singleton clauses, and $\hat{C}$ the rest of the clauses. Now, one constructs indicator random variables X_c for $c \in C \cup \hat{C}$ as follows. We define $X_c = 1$ if clause c is satisfied, and $X_c = 0$ otherwise. Now for each variable x_i, assign $x_i = 1$ with .5 probability, and $x_i = 0$ with .5 probability. Now, each singleton clause $c \in C$ is satisfied with probability .5. Then, one calculates the probability that some clause $c \in \hat{C}$ fails to be satisfied, if c has length k, this happens with probability $\frac{1}{2^k}$ where $k \geq 2$. So, c is satisfied with probability $1 - \frac{1}{2^k} \geq 1 - \frac{1}{2^2} = \frac{3}{4}$. So, for any $c \in \hat{C}$, c is satisfied with probability $\geq \frac{3}{4}$. Now, we can calculate the expected number of satisfied clauses.

$$\begin{aligned} E[\sum_{c \in C} X_c + \sum_{c \in \hat{C}} X_c] &= \sum_{c \in C} E[X_c] + \sum_{c \in \hat{C}} E[X_c] \\ &\geq \sum_{c \in C} \frac{1}{2} + \sum_{c \in \hat{C}} \frac{3}{4} \end{aligned}$$

Now, either $|C| \geq \frac{3}{5}m$ or $|C| < \frac{3}{5}m$. If $|C| \geq \frac{3}{5}$, then one can satisfy them all, which then gives that at least $\frac{3}{5}m = 0.6m$ clauses are satisfied. Otherwise, $|C| < \frac{3}{5}m$, which implies that $|\hat{C}| = m - |C| \geq m - \frac{3}{5}m = \frac{2}{5}m$. So, returning to our calculation, we get

$$\begin{aligned} E[\sum_{c \in C} X_c + \sum_{c \in \hat{C}} X_c] &\geq \sum_{c \in C} \frac{1}{2} + \sum_{c \in \hat{C}} \frac{3}{4} \\ &= \frac{1}{2}|C| + \frac{3}{4}|\hat{C}| \\ &= \frac{1}{2}|C| + \frac{3}{4}(m - |C|) \\ &= \frac{1}{2}|C| + \frac{3}{4}m - \frac{3}{4}|C| \\ &= \frac{3}{4}m - \frac{1}{4}|C| \\ &\geq \frac{3}{4}m - \frac{1}{4}\frac{3}{5}m \\ &= (\frac{3}{4} - \frac{3}{20})m = \frac{6}{10}m. \end{aligned}$$

Now, since for a random assignment the expected number of satisfied clauses is ≥ 0.6, we know that there exists an assignment that achieves at least the expectation and we are done.

Problem 4 Matrix Multiplication Checking

Let A, B and C be $n \times n$ real matrices. Construct a multivariate polynomial Q such that $Q = 0$ if and only if $AB = C$, and that evaluating Q at a point takes $O(n^2)$ time. (Remark: this reduces checking $AB = C$ to testing polynomial identities.)

Let $x = (x_1, x_2, \ldots, x_n)^T \in \mathbb{R}^n$ be a vector. Then, $AB = C$ if and only if $ABx = Cx$ for all $x \in \mathbb{R}^n$. Then, one notes that $ABx = Cx$ if and only if $ABx - Cx = 0 \in \mathbb{R}^n$. We then expand

$$
\begin{aligned}
A(Bx) - Cx &= A \begin{bmatrix} \sum_{j=1}^n B_j^1 x_j \\ \sum_{j=1}^n B_j^2 x_j \\ \vdots \\ \sum_{j=1}^n B_j^n x_j \end{bmatrix} - \begin{bmatrix} \sum_{j=1}^n C_j^1 x_j \\ \sum_{j=1}^n C_j^2 x_j \\ \vdots \\ \sum_{j=1}^n C_j^n x_j \end{bmatrix} \\
&= \begin{bmatrix} \sum_{k=1}^n (A_k^1 (\sum_{j=1}^n B_j^k x_j)) \\ \sum_{k=1}^n (A_k^2 (\sum_{j=1}^n B_j^k x_j)) \\ \vdots \\ \sum_{k=1}^n (A_k^n (\sum_{j=1}^n B_j^k x_j)) \end{bmatrix} - \begin{bmatrix} \sum_{k=1}^n C_k^1 x_k \\ \sum_{j=k}^n C_k^2 x_k \\ \vdots \\ \sum_{j=k}^n C_k^n x_k \end{bmatrix} \\
&= \begin{bmatrix} \sum_{k=1}^n (A_k^1 (\sum_{j=1}^n B_j^k x_j)) - \sum_{k=1}^n C_k^1 x_k \\ \sum_{k=1}^n (A_k^2 (\sum_{j=1}^n B_j^k x_j)) - \sum_{k=1}^n C_k^2 x_k \\ \vdots \\ \sum_{k=1}^n (A_k^n (\sum_{j=1}^n B_j^k x_j)) - \sum_{k=1}^n C_k^n x_k \end{bmatrix} \\
&= \begin{bmatrix} \sum_{k=1}^n ((A_k^1 (\sum_{j=1}^n B_j^k x_j)) - C_k^1 x_k) \\ \sum_{k=1}^n ((A_k^2 (\sum_{j=1}^n B_j^k x_j)) - C_k^2 x_k) \\ \vdots \\ \sum_{k=1}^n ((A_k^n (\sum_{j=1}^n B_j^k x_j)) - C_k^n x_k) \end{bmatrix}
\end{aligned}
$$

One then notes that $ABx - Cx = 0$ if and only if its squared (usual L_2) norm is 0. Namely, $AB = C$ if and only if the following polynomial is identically the zero polynomial.

$$
\sum_{L=1}^n \left(\left(\sum_{k=1}^n \left((A_k^L (\sum_{j=1}^n B_j^k x_j)) - C_k^L x_k \right) \right)^2 \right) \equiv 0
$$

Equivalently, $AB = C$ if and only if

$$
f(x_1, x_2, \ldots, x_n) = \sum_{L=1}^n \left(\left(\sum_{k=1}^n \left((A_k^L (\sum_{j=1}^n B_j^k x_j)) - C_k^L x_k \right) \right)^2 \right) = 0 \text{ for all } \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \in \mathbb{R}^n.
$$

How long does it take to evaluate $f(x_1, x_2, \ldots, x_n)$? It takes $O(n^2)$ time. Why? Computing

$$
Bx = \begin{bmatrix} \sum_{j=1}^n B_j^1 x_j \\ \sum_{j=1}^n B_j^2 x_j \\ \vdots \\ \sum_{j=1}^n B_j^n x_j \end{bmatrix}
$$

take $O(n^2)$ time. Then, computing

$$A(Bx) = \begin{bmatrix} \sum_{k=1}^{n}(A_k^1(\sum_{j=1}^{n} B_j^k x_j)) \\ \sum_{k=1}^{n}(A_k^2(\sum_{j=1}^{n} B_j^k x_j)) \\ \vdots \\ \sum_{k=1}^{n}(A_k^n(\sum_{j=1}^{n} B_j^k x_j)) \end{bmatrix}$$

using the column vector Bx takes $O(n^2)$ time. Then, computing

$$Cx = \begin{bmatrix} \sum_{k=1}^{n} C_k^1 x_k \\ \sum_{j=k}^{n} C_k^2 x_k \\ \vdots \\ \sum_{j=k}^{n} C_k^n x_k \end{bmatrix}$$

takes $O(n^2)$ time. Then, subtracting $ABx - Cx$ take $O(n)$ time. Finally, squaring each entry of $ABx - Cx$ takes $O(n)$ time. Finally, summing all n squared entries takes $O(n)$ time. So, overall the time complexity is $O(n^2)$.

Problem 5 Tutte's Theorem

Consider an arbitrary (possibly non-bipartite) graph $G = (V, E)$ where $V = \{v_1, \ldots, v_n\}$. A skew-symmetric matrix A is defined to be a matrix in which for all i and j, $A_{ij} = -A_{ji}$. Let A be the $n \times n$ skew-symmetric matrix obtained from $G = (V, E)$ as follows. A distinct indeterminate x_{ij} is associated with the edge (v_i, v_j) where $i < j$, and the corresponding matrix entries are given by $A_{ij} = x_{ij}$ and $A_{ji} = -x_{ij}$; more succinctly,

$$A_{ij} = \begin{cases} x_{ij}, & (v_i, v_j) \in E \text{ and } i < j; \\ -x_{ji}, & (v_i, v_j) \in E \text{ and } i > j; \\ 0, & \text{otherwise.} \end{cases}$$

This matrix is called the Tutte matrix of the graph G. The determinant of A, denoted by $\det(A)$, is a multivariate polynomial in x_{ij}'s. Show that G has a perfect matching if and only if $\det(A) \neq 0$.

Say that G has a perfect matching. This happens exactly when intuitively there is a permutation matrix sitting inside $G's$ adjacency matrix M. Recall that a permutation matrix is a $0, 1$ matrix in which there is exactly one 1 in each row and column. Translated to the language of graphs this corresponds to a subset of unordered pairs of vertices (i.e. edges) (which correspond to the entries set to 1 in our permutation matrix) which are pairwise disjoint. They are pairwise disjoint because there is only one 1 (i.e. chosen edge) in the row or column corresponding to any vertex. More precisely, G has a perfect matching exactly when for some permutation $\sigma \in S_n$ (with permutation matrix defined by $N^i_{\sigma j} = 1$ if $\sigma(i) = j$ and $N^i_{\sigma j} = 0$ otherwise) the matrix difference $M - N_\sigma$ has all non-negative entries. Now, we wish to show that there exists such a σ exactly when $\det(A) \neq 0$. First, say there exists such a $\sigma \in S_n$. Then, namely, setting the variables $x_{ij} = 1$ when $j = \sigma(i)$ and setting all other variables in A to 0 results in a matrix with determinant 1 or -1 (because such a matrix is a permutation matrix, N_σ, and any permutation matrix can be obtained from the identity matrix by successively swapping rows which either multiplies the determinant by 1 or -1). So, what we have done is evaluated our multivariate polynomial $\det(A)$ at a specific point (set of values for the variables x_{ij}). Specifically, we see that $\det(A)(x_{i\sigma(i)} = 1$ if $\sigma(i) > i$, $x_{i\sigma(i)} = -1$ if $\sigma(i) < i$, and $x_{ij} = 0$ otherwise$) = \pm\det(N_\sigma) = \pm 1 \neq 0$, which means that $\det(A)$ is not the zero function (note that here the matrix $A(x_{i\sigma(i)} = 1$ if $\sigma(i) > i$, $x_{i\sigma(i)} = -1$ if $\sigma(i) < i$, and $x_{ij} = 0$ otherwise$)$ has determinant $\pm\det(N_\sigma)$ because $\det(A)(x_{i\sigma(i)} = 1$ if $\sigma(i) > i$, $x_{i\sigma(i)} = -1$ if $\sigma(i) < i$, and $x_{ij} = 0$ otherwise$) = (-1)^r \det(N_\sigma)$ where $r = \#\{i | \sigma(i) < i\}$ since the corresponding rows of N_σ have been multiplied by -1 to get $A(x_{i\sigma(i)} = 1$ if $\sigma(i) > i$, $x_{i\sigma(i)} = -1$ if $\sigma(i) < i$, and $x_{ij} = 0$ otherwise$)$). Now, we wish to prove the other direction. Namely, we wish to show that if $\det(A) \neq 0$ (as a function), then G has a perfect matching. Well, $\det(A) \neq 0$ means that there is a some assignment to the variables x_{ij} such that $\det(A)$ evaluated at that point is not zero. Well, A^i_j skew symmetric implies that

$$\begin{aligned} \det(A(x)) &= \det((A(x))^T) \\ &= \det(-A(x)) \\ &= (-1)^n \det(A(x)) \text{ for all } x \end{aligned}$$

So, if n is odd, then one has that $-\det(A(x)) = \det(A(x))$ for all x, which implies that $\det(A(x)) = 0$ for all x, a contradiction. So, n is even. That's good news! That means there is hope of finding a perfect matching. So, at some x we have that $\det(A(x)) \neq 0$ implies that no row or column is entirely zero. We then note that A skew symmetric or namely $A^T = -A$ implies that

$$A^2 = (-A^T)^2 = (A^T)^2 = (A^2)^T.$$

So, in other words A^2 is symmetric. Now, $\det(A(x)) \neq 0$ for some x means that

$$\det((A(x))^2) = (\det(A(x)))^2 \neq 0.$$

Let's actually calculate $(A(x))^2$.

$$(A(x))^2 = \begin{bmatrix} \sum_{k=1}^n A_k^1 A_j^k \\ \sum_{k=1}^n A_k^2 A_j^k \\ \vdots \\ \sum_{k=1}^n A_k^n A_j^k \end{bmatrix}$$

Note that here each row actually has n entries, one for each $j \in [n]$ as you can note that j has not been summed out.
Then, note that $\det((A(x))^2) \neq 0$ for some x implies that $(A(x))^2$ has full row and column rank. In particular, this means that there is at least 1 non-zero entry in each row and column of $(A(x))^2$. Now, say one wants to pick out a subset of the non-zero entries of $(A(x))^2 =: B$ so that there is exactly one non-zero entry in each row and column. We show that this is possible by induction on the number of rows in B. First, say B is a 1 by 1 matrix. Then, it's only entry is non-zero (since its determinant is non-zero), so choose B_1^1 and we are done. Now, for the inductive step, one first determines if there is any row (or column) index $r \in [n]$ of B such that the row (or same thing with columns, but B symmetric means that there is a row with exactly one non-zero entry exactly when there is some column with exactly one non-zero entry) B^r has exactly one non-zero entry. If so one chooses this entry $B_{c_r}^r$ as the non-zero entry in this row and column. Then, one deletes row r and column c_r from B to get B'. Now, provided that B' still satisfies determinant non-zero one can apply the inductive hypothesis to B' to get subset of the non-zero entries of B' such that for any row or column, exactly one of these entries belongs to it, and then combining this set with $B_{c_r}^r$ gives a set of n entries $\{B_{j_i}^i : i \in [n], j_i \in [n], B_{j_i}^i \neq 0 \forall i, \text{ and } j_r \neq j_s \text{ for all } r \neq s\}$. One notes that the determinant of B' is non-zero by noting that $\det(B) = \pm\det(B')$ by expanding along row r. So, that case is done. Now, suppose all rows and columns of B have more than one non-zero entry. Then, one does the same thing, one picks any row, say WLOG B^1 and chooses a non-zero entry $B_{j_1}^1 \neq 0$, then deletes row 1 and column j_1 of B to get B'. Now, one wants to ensure that B' still has a non-zero determinant or still has full rank in order to apply the inductive hypothesis. Indeed it does have full rank because, in B, there was more than one non-zero entry in each row. So, for any row $r \in \{2, \dots, n\}$ $B'^{r-1} = B^r$ has some entry $B'^{r-1}_{j_{r-1}} = B_{k_r}^r \neq 0$ (where k_r is the column of B corresponding to the column j_{r-1} of B', recall that we deleted a column of B to get B') WITH $k_r \neq j_1$ because namely, $B_{j_1}^r$ may have been a non-zero entry of $B^r = B'^{r-1}$ that has been deleted but there is another non-zero entry $B_{k_r}^r$ in row r, so there is still at least one non-zero entry in each row of B'. To see that each column of B' still has at least one non-zero entry one notes that by assumption of this case, every column of B had at least 2 non-zero entries. So, take a column $B'_c = B_k$ of B' (which is the kth column of B), we wish to show there is some $r \in \{2, \dots, n\}$ such that $B'^{r-1}_c = B_k^r \neq 0$. However, such an r cannot equal 1 because that row has been deleted. Though now, we recall that the kth column of B had at least 2 non-zero entries, so even if one was B_k^1 which has been deleted, there is some other non-zero entry B_c^r in this column with $r \neq 1$, and now we have shown $\det(B') \neq 0$, which means that we can apply the inductive hypothesis and we are done.

So, one can choose a set of non-zero entries $\{B_{j_i}^i : i \in [n], j_i \in [n], B_{j_i}^i \neq 0 \forall i, \text{ and } j_r \neq j_s \text{ for all } r \neq s\}$ of B with exactly one of these entries in any given row or column of B.

Problem 6 Maximum Matching

The Maximum-Matching problem is as follows: Given a graph G and an integer $k \geq 1$, decide whether there is a matching in G of size at least k. Given a randomized algorithm for testing the existence of a perfect matching in a graph, describe how we can use this to solve the Maximum-Matching problem.

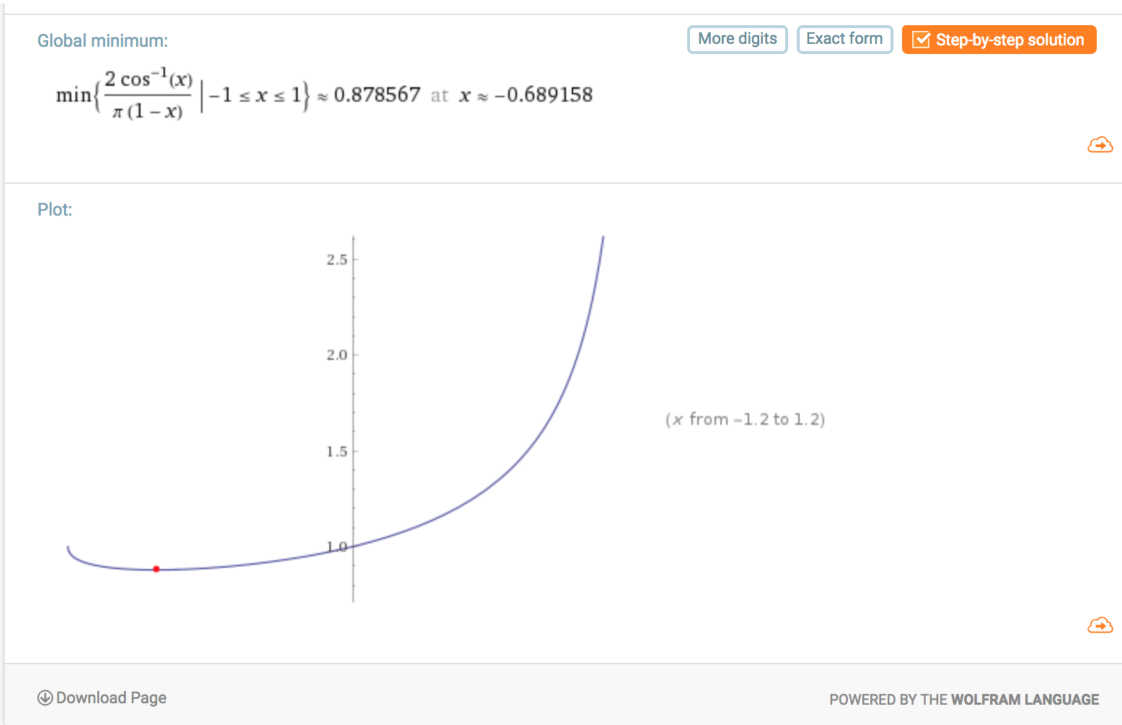
Take all subsets $U \subset V(G)$ with $|U| = 2k$. Then, check if $G[U]$ has a perfect matching. If so, G has a matching of size at least k.

(Bonus) Problem 7 Plot for Fun

Plot the graph of the function

$$f(x) = \frac{2\arccos(x)}{\pi(1-x)}$$

for $x \in [-1, 1]$ and numerically find the minimum value of f. (You don't need to give any proof or algorithm.)



Define

$$\sum_{C_i \in X_2} X_i = \text{ the number of satisfied clauses}$$

where $Y_i = 1$ if and only if C_i is satisfied and is 0 otherwise.
Then,

$$E[\sum_{C_i \in X_2} Y_i] = \sum_{C_i \in X_2} E[Y_i].$$

However, it is not true always that $E[Y_i] = \frac{3}{4}$. Namely, the values of various $Y_i's$ are not mutually independent, so you would need to sum over various conditioned values like this