Selecting Chromosomes for Polygenic Traits

Or Zuk

Department of Statistics and Data Science
The Hebrew University of Jerusalem,
Jerusalem 91905, Israel
or.zuk@mail.huji.ac.il

Abstract. We define and study the problem of *chromosomal selection* for multiple complex traits. In this problem, it is assumed that one can construct a genome by selecting different genomic parts (e.g. chromosomes) from different cells. The constructed genome is associated with a vector of polygenic scores, obtained by summing the polygenic scores of the different genomic parts, and the goal is to minimize a loss function of this vector. While out of reach today, the problem may become relevant in the future with emerging future technologies, and may yield far greater gains in the loss compared to the present day technology of as embryo selection, provided that technological and ethical barriers are overcome. We suggest and study several natural loss functions relevant for both quantitative traits and disease. We propose two algorithms, a Branch-and-Bound technique, to solve the problem for multiple traits and any monotone loss function, and a convex relaxation algorithm applicable for any differentiable loss. Finally, we use the infinitesimal model for genetic architecture to approximate the potential gain achieved by chromosomal selection for multiple traits.

Keywords: Chromosome selection · Polygenic Scores · Branch and Bound · Relaxation

1 Introduction

Polygenic Scores (PS) are genetic scores predicting a phenotype of interest, by combining the contribution of multiple genomic alleles. In the last few years hundreds of polygenic scores were developed for predicting complex diseases and quantitative traits in humans [25], with the scores coefficients typically fitted in large Genome-Wide-Association-Studies (GWAS) [22, 39]. The accuracy of PSs is expected to improve significantly in the upcoming years due to increase in GWAS sample sizes, inclusion of additional populations [32], usage of whole genome sequencing (rather than genotyping using SNP-arrays) that enable to profile of additional (in particular rare) variants [14], and improvement in statistical methods for fitting such scores [5, 11, 29].

These recent advances make it possible to screen embryos for common, complex conditions and traits, when using in vitro fertilization (IVF), in a technology termed Polygenic Embryo Screening (PES). Conceptually, PES is quite straightforward: screen the potential embryos to calculate their PS for traits of interest, and select the embryo maximizing a PS (e.g. minimizing disease risk), and the technology is already offered commercially and is in use in the clinic [35]. Several works have analyzed the potential benefits and risks using theoretical and empirical analysis but the effectiveness of current technology is debatable [21, 26, 35, 36]. In particular, the limited number of embryos to select for (typically not more than $5-7$) may limit the potential benefits from the technology, especially when selecting for multiple traits of interest simultaneously [26].

With novel technological advances, it might be possible in the future to go beyond embryo selection, and select and combine parts of the genome of different cells. Such flexibility, is made possible, will lead to a far greater space of possibilities for selection compared to PES, with potentially significantly larger benefits in disease risk reduction. For example, chromosomal transplantation was recently demonstrated in vitro [30, 31], in order to replace a defective chromosome by a normal one. Similar technologies are used in the lab to study humanized animal models [37, 41]. For complex polygenic traits, selection of individual chromosomes or large-scale genomic regions from available cells may be more effective compared to genome engineering approaches gene editing using the CRISPR-CAS9 system [18] that affect only a one or a handful of genes.

A major technological challenge will be to determine the PS of individual cells and chromosomes in a non-invasive manner. Such methods may be available for oocytes [17]. Alternatively, phenotyping individual cells (e.g. using imaging techniques) [20] may be correlated with DNA quality [28], and may provide indirect proxy for polygenic scores. Assuming that technological and ethical issues are resolved, allowing chromosomal selection in humans, animal models or agriculture, computational methods and statistical analysis are are needed in order to fully utilize the potential of the different selection methods. The goal of this paper is to develop these methods and analysis. Specifically, we focus on *chromosomal selection* for multiple quantitative traits and diseases. In this problem, multiple copies are available for each chromosome (or possibly a smaller genomic part), and based on these copies' PS we select one of them. Such choices are utilized to generate an embryo from the different chromosomal parts. We address the following two main questions:

1. How should the chromosomes be selected in order to maximize utility across multiple traits? When selecting for T traits, each selection $\mathbf{c}$ of chromosomes leads to an overall genomic score

vector $\boldsymbol{X}_{\mathbf{c}} \in \mathbb{R}^T$. A loss function $\mathcal{L} : \mathbb{R}^T \to \mathbb{R}$ is defined and our goal is to find the selection $\mathbf{c}$ minimizing the loss $\mathcal{L}(\boldsymbol{X}_{\mathbf{c}})$, and compare it to the loss obtained for random selection. When C copies are available for each of the M chromosomes, the total number of possible selections C^M is exponential in M, hence the need to design efficient general algorithms for the problem.

2. What is the expected gain when- using optimal chromosomal selection? how does it compare to the baseline, i.e. selecting embryos at random, as well as to the embryo selection procedure that is enabled by current technology and already offered to patients [26, 35]?

Our main contributions are threefold: first, we formulate the chromosomal selection problem mathematically. Second, we provide two algorithms for chromosomal selection for multiple traits and general loss functions and investigate their empirical performance. Third, we estimate the expected gain achieved by chromosome selection, for both linear loss functions where we establish an analytic approximate formula, as well as for a few nonlinear loss functions using simulations.

2 The Chromosomal Selection Problem

2.1 Background and Selection Problems for Polygenic Scores

Consider a genome composed of M distinct chromosomes, where for diploid cells we count the maternal and paternal chromosomes separately, hence for example for a human diploid cell $M = 45$ with 22 pairs of autosomes and the 'XX' pair (as explained later, in most plausible scenarios selection for the 'XY' pair is determined by the sex and not the scores). We associate with each chromosome a Polygenic Score (PS) vector representing the genetic contribution of the chromosome to a T traits of interest. Suppose that we have C distinct cells, each with its own genome, hence C copies are available for each chromosome overall. Our goal is to select one copy for each chromosome, possibly under constraints, yielding a full genome with desired properties in terms of the resulting polygenic score. For example, in *embryo selection*, the C cells are C embryos obtained from the same parents, and the selection is performed by simply choosing one of the cells, such that all selected chromosomes belong to the same cell. In *chromosomal selection*, it may be possible to select different chromosomes from different cells. For example, suppose that a diploid parental cell and a diploid maternal cell are available, and both are reprogrammed to create haploid sperm and oocytes cells, respectively [7, 16], and later to yield a viable (diploid) embryo after fertilization. Suppose further that it will be possible to select the maternal or paternal copy independently for each chromosomes in the created haploid cells. Hence, there are overall $C = 2$ copies of each chromosome, with $\frac{M}{2}$ chromosomes in each haploid cell (ignoring for simplicity the uniqueness of the sex chromosomes), and a space of 2^M overall possible resulting genomes from the embryo, depending on the selected copy at each chromosome. Figure 1(b) shows an illustration of chromosomal selection for sperm cells and oocytes. A simplified variant of the problem is obtained when an oocyte is already available, and we only select chromosomes from a diploid parental cell for sperm cell, or vice versa, hence $M = 22$ or 23, and the scores vector of the selected gamete can be then added to the scores of the available gamete (assumed w.l.o.g. to be $\mathbf{0}_T$), yielding a chromosomal selection problem with a smaller M value. Additional scenarios in which chromosomal selection may be possible are described in Appendix Section C.

2.2 Preliminaries

We first introduce mathematical notations used throughout the paper. For a natural number $n \in \mathbb{N}$ denote the set $\{1, .., n\}$ by $[n]$. For two natural numbers $m \leq n$ denote the set $\{m, m+1, .., n\}$ by $[m, n]$. The vector of all zeros (ones) of length n is denoted $\mathbf{0}_n$ ($\mathbf{1}_n$).

For a polytope $\Delta \in \mathbb{R}^n$, we define the projection operator $\mathbb{P}_\Delta(x) \equiv \underset{y \in \Delta}{argmin} ||y - x||^2$.

Tensors Let $\boldsymbol{X} \in \mathbb{R}^{m \times n \times p}$ be a 3^{rd} order tensor with elements X_{ijk}. We use the $\bullet$ notation to define lower-dimensional fibers of a tensor. For example, $\boldsymbol{X}_{ij\bullet}$ denotes the vector $(x_{ij1}, .., x_{ijp}) \in \mathbb{R}^p$. Similarly, $\boldsymbol{X}_{\bullet j\bullet}$ is a matrix of size $m \times p$ containing all elements $X_{ijk}, \forall i \in [m], \forall k \in [p]$. We can also describe sub-tensors obtained by taking subsets of the indices across each dimension. For example, $X_{[i][j]\bullet}$ is a 3^{rd} order tensor of size $i \times j \times p$ obtained by taking the first i and j coordinates, on the first and second dimension, respectively.

For a 3^{rd}-order tensor $\boldsymbol{X} \in \mathbb{R}^{m \times n \times p}$ and a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, define the *2-mode* tensor-by-matrix product [1, 23], as a matrix $[\boldsymbol{X} \times_2 \mathbf{A}] \in \mathbb{R}_{m \times p}$, with elements defined by:

$$[\boldsymbol{X} \times_2 \mathbf{A}]_{ik} = \sum_{j=1}^{n} X_{ijk} A_{ij}, \ \forall i \in [m], \forall k \in [p]. \tag{1}$$

Gaussian Distribution For the multivariate Gaussian distribution with mean μ and variance Σ, denote by $\phi(\mathbf{x}; \mu, \Sigma)$ and $\Phi(\mathbf{x}; \mu, \Sigma)$ the density function and cumulative distribution function, respectively. When μ, Σ are omitted, Φ and ϕ refer to the standard multivariate Gaussian distribution with mean zero and identity covariance matrix. For a measurable set $\mathcal{A} \subset \mathbb{R}^T$ denote by $\Phi(\mathcal{A}; \mu, \Sigma)$ the probability of the set under the Gaussian distribution, i.e. $\Phi(\mathcal{A}; \mu, \Sigma) \equiv \int_{\mathcal{A}} \phi(\mathbf{x}; \mu, \Sigma) d\mathbf{x}$.

2.3 Chromosome selection for Multiple Traits

Let $\boldsymbol{X} \in \mathbb{R}^{M \times C \times T}$ be a 3^{rd} order tensor of polygenic scores, where X_{ijk} denotes the score in chromosome i of copy j for trait k. Let $\mathbf{c} = (c_1, .., c_M) \in \{1, .., C\}^M$ be a selection vector. The associated selected polygenic vector is defined as $\boldsymbol{X}_{\mathbf{c}} \equiv \sum_{i=1}^{M} X_{ic_i\bullet} \in \mathbb{R}^T$, with $\boldsymbol{X}_{\mathbf{c}_k}$ denoting its k-th element $(\boldsymbol{X}_{\mathbf{c}})_k$, $\forall k \in [T]$. Our goal is to find the selected score vector $\boldsymbol{X}_{\mathbf{c}}$ minimizing a loss function of our choice. The *multi-trait chromosomal selection problem* is defined as follows:

Problem 1. Given a 3^{rd} order tensor of scores $\boldsymbol{X} \in \mathbb{R}^{M \times C \times T}$, and a loss function $\mathcal{L} : \mathbb{R}^T \to \mathbb{R}$, find a vector $\mathbf{c}^* \in \{1, .., C\}^M$ minimizing the loss: $\mathbf{c}^* \equiv \underset{\mathbf{c} \in \{1, .., C\}^M}{argmin} \ \mathcal{L}(\boldsymbol{X}_{\mathbf{c}})$.

Table 1 lists several examples of natural loss functions of interest for both quantitative and disease traits. The computational difficulty of Problem 1 above depends on the loss function $\mathcal{L}$. For example, when the $\mathcal{L}$ is linear in the polygenic risk vector $\boldsymbol{X}_{\mathbf{c}}$, we can select the optimal vector for each of the M independently, and the computational problem becomes trivial. This is the case when selecting for maximizing a weighted combinations of quantitative traits. However, for non-linear loss functions (e.g. minimizing the overall disease probability over multiple diseases),

selecting the best chromosome may be computationally challenging since we need to take into account the scores of all chromosomes jointly. In Section 4, we propose two algorithms for finding the optimal selection applicable for general classes of loss functions.

In particular, many natural loss functions are *monotone* functions of the scores vector. This montonicity may be used when solving Problem 1. Formally, we define monotone loss functions with respect to the (partial) product order between vectors as follows:

Definition 1. *A vector* $\mathbf{y} \in \mathbb{R}^n$ *dominates a vector* $\mathbf{x} \in \mathbb{R}^n$*, denoted* $\mathbf{x} \prec \mathbf{y}$ *if* $x_i < y_i$, $\forall i \in [n]$*. We say that a loss function* $\mathcal{L} : \mathbb{R}^n \to \mathbb{R}$ *is monotonically non-increasing if for any two vectors* $\boldsymbol{X}_\mathbf{c} \prec \mathbf{Y}_\mathbf{c}$ *we have* $\mathcal{L}(\boldsymbol{X}_\mathbf{c}) \geq \mathcal{L}(\mathbf{Y}_\mathbf{c})$*.*

Loss	Formula	Monotone?	Convex?	Algorithms
Linear	$\sum_{i=1}^{T} w_i \boldsymbol{X}_{\mathbf{c}_i}$	YES	YES	Simple Selection
Stabilizing	$\sum_{i=1}^{T} w_i \boldsymbol{X}^2_{\mathbf{c}_i}$	NO	YES	Relaxation
Disease-linear	$\sum_{i=1}^{T} w_i P(D_i = 1 \vert \boldsymbol{X}_\mathbf{c})$	YES	NO	Branch-and-Bound/Relaxation
Disease-free	$-P(\mathbf{D} = \mathbf{0}_T \vert \boldsymbol{X}_\mathbf{c})$	YES	NO	Branch-and-Bound/Relaxation

Table 1: Example loss functions $\mathcal{L}(\boldsymbol{X}_\mathbf{c})$ and their properties. (i) For quantitative traits we define a linear loss, with a weight vector $\mathbf{w} \in \mathbb{R}^T$ representing the importance of each trait, and where $\boldsymbol{X}_{\mathbf{c}_i}$ denotes the i-th element of the vector $\boldsymbol{X}_\mathbf{c}$. (ii) In *stabilizing selection* [15, 33, 34], we select such that the resulting scores will be as close as possible to the mean (assumed to be zero w.l.o.g). This selection is desired when assuming that the value of a quantitative trait is already distributed around its optimum at equilibrium, and significant deviations from it may be harmful, and are penalized by a quadratic term. (iii) When the polygenic score vector determines the probability of several diseases, it is desirable to select such that the disease probabilities will be minimized. Consider T diseases with a vector of prevalences $\mathbf{K} \in [0,1]^T$. The binary vector $\mathbf{D} \in \{0,1\}^T$ represents the status of each disease, where $P(D_i = 1 \vert \boldsymbol{X}_\mathbf{c})$ is defined according to the liability threshold model [9]. See details about the statistical model relating the score and other genetic and environmental contributions to the vector of phenotypes in the Appendix, Section B.1. (iv) A loss can represent the probability of being disease free for multiple diseases simultaneously.

The Gain due to Selection

Definition 2. *For a tensor* $\boldsymbol{X}$ *of scores, and a loss function* $\mathcal{L}$*, we define the* Gain G *due to chromosomal selection and* Gain G_e *due to embryo selection as the differences between the optimal loss and the expected loss* $\mathbb{E}_c$ *when selecting at random, i.e. with respect to the uniform distribution over all* C^M *possible choices of chromosomes:*

$$G(\boldsymbol{X}; \mathcal{L}) \equiv \mathbb{E}_\mathbf{c} \mathcal{L}(\boldsymbol{X}_\mathbf{c}) - \mathcal{L}(\boldsymbol{X}_{\mathbf{c}^*})$$
$$G_e(\boldsymbol{X}; \mathcal{L}) \equiv \mathbb{E}_\mathbf{c} \mathcal{L}(\boldsymbol{X}_\mathbf{c}) - \min_{j \in [C]} \mathcal{L}(\sum_{i=1}^{M} \boldsymbol{X}_{ij\bullet}). \tag{2}$$

The gain $G_e(\boldsymbol{X};\mathcal{L})$ is similar in spirit to previous definitions given in [21, 26], but with two major differences: First, it is defined for a general loss whereas [21, 26] defined the gain only for additive losses for quantitative and disease traits. Second, the gain in [21, 26] was defined with respect to the actual trait value, that is determined by the score, as well as non-score genetic and environmental components. Here, the gain and the loss are defined in terms of the scores only, hence the gain can be viewed as expectation over a latent variable representing the phenotype value of the previous gain. By definition, $G_e(\boldsymbol{X};\mathcal{L}) \leq G(\boldsymbol{X};\mathcal{L})$ and we are interested in the expected gain of both approaches compared to random selection, and their expected (non-negative) difference. In Section B we use a statistical model for the scores to get approximate formulas for the expected gain and its dependence on model parameters for the linear loss.

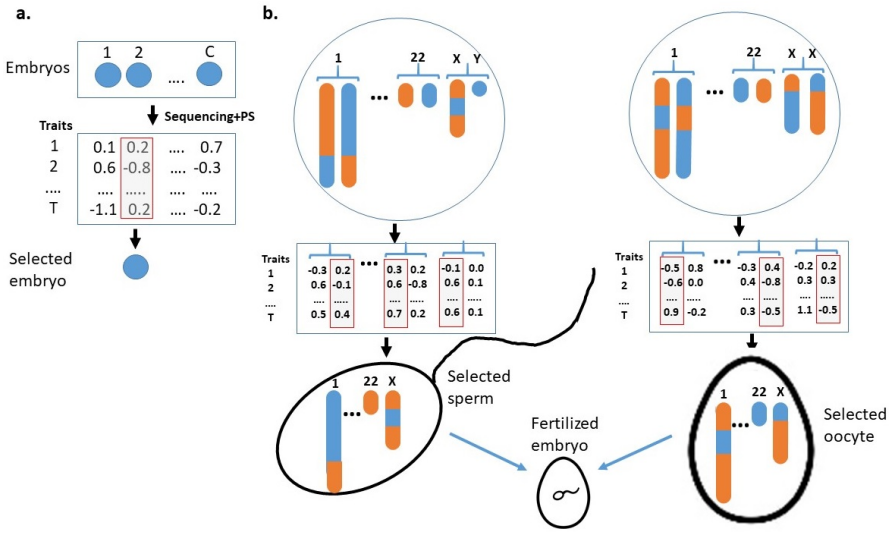

Fig. 1: Illustration of different types of selection based of polygenic scores for T traits. (a.) Embryo selection out of n viable embryos is performed by selecting an embryo based on its PS after fertilization and growing the embryos. (b.) Chromosomal selection with one sperm cell and one oocyte is performed by selecting for each chromosome one of the $C = 2$ copies and creating a fertilized embryo using all the selected chromosomes.

3 The Expected Gain

The gain $G = G(\boldsymbol{X};\mathcal{L})$ defined in eq. (2) represents the utility of chromosomal selection for a concrete set of chromosomes with scores. We are interested in statistical properties of the gain in a population, hence the need for a statistical model for the scores tensor $\boldsymbol{X}$. That is, suppose that $\boldsymbol{X} \sim P_X$. Hence $G(\boldsymbol{X};\mathcal{L})$ is also a random variable determined by the scores distribution P_X and the loss $\mathcal{L}$. We study here *expected gain* $\mathbb{E}_{\boldsymbol{X}} G(\boldsymbol{X};\mathcal{L})$, and similarly the expected gain due to embryo selection $\mathbb{E}_{\boldsymbol{X}} G_e(\boldsymbol{X};\mathcal{L})$. In Section B.1 we derive a statistical model for $\boldsymbol{X}$ based on

quantitative genetics principles, extending the models for whole-genome scores in [21, 26]. Under this model, the scores for the different chromosomes are independent, and the covariance matrix of a randomly selected vector $\boldsymbol{X}_c$ is denoted by $\boldsymbol{\Sigma}^{(\boldsymbol{X})}$. For the linear loss $\mathcal{L} = \mathbf{w}^t \boldsymbol{X}_c$, we showed in [21] that the gain due to embryo selection is

$$\mathbb{E}_{\boldsymbol{X}} G_e \equiv \mathbb{E}_{\boldsymbol{X}}[\max_{i=1}^{C}(\mathbf{w}^t \boldsymbol{X}_{i\bullet}^{(e)})] \approx 0.77\sqrt{\mathbf{w}^t \boldsymbol{\Sigma}^{(\boldsymbol{X})}\mathbf{w}}\sqrt{\log C}. \tag{3}$$

Chromosomal selection is simple for this case and is achieved by selecting for each chromosome i the copy j minimizing $\mathbf{w}^t X_{ij\bullet}$. Using this property, we derive the approximate gain for chromosomal selection as (see details in Appendix B.1):

$$\mathbb{E}_{\boldsymbol{X}} G \approx \Big(\sum_{k=1}^{M} \alpha_i\Big) 0.77\sqrt{\mathbf{w}^t \boldsymbol{\Sigma}^{(\boldsymbol{X})}\mathbf{w}}\sqrt{\log C} \tag{4}$$

where α_i^2 is the proportion of score variance explained by chromosome i, satisfying $\sum_{i=1}^{M} \alpha_i^2 = 1$. The expected gain for chromosomal selection is thus roughly $\sum\limits_{i=1}^{M} \alpha_i$-fold higher compared to the expected gain for embryo selection in eq. (3). For the α_i values in Table 2 in the Appendix representing human chromosomal lengths, this gives a 4.68-fold improvement. For general (non-linear) loss functions, we compute the expected gain numerically using simulations, as is shown in Section 5.

4 Algorithms for Chromosome Selection

The optimization problem 1 is difficult due to the exponential search space of size C^M. For example, selecting for 23 chromosomes in each of a single sperm and oocyte cell in humans, the number of possible selections is $2^{45} \approx 3 \times 10^{13}$. We describe next two classes of algorithms for the problem: a Branch-and-Bound that eliminates dominated selections approach, and a relaxation of the discrete selector variables c_i to continuous vectors in the simplex. The two algorithms are applicable for different scenarios: The Branch-and-Bound techniques can be applied to any monotone loss, are guaranteed to yield an optimal solution, but their worst-case computational complexity is exponential in M. The Relaxation approach is polynomial and can be applied to any differentiable loss, but has no optimality guarantees.

4.1 A Branch-and-Bound algorithm

In the Branch-and-Bound algorithm for selecting chromosomes *for monotone loss functions*, we grow a tree of all possible selected chromosomes, and at each level keep only leafs not dominated by other leafs. Finally, we evaluate the loss of all leafs at the last level. A tree of depth b is represented as a collection of paths from the root to the leafs $\Gamma = \{\mathbf{c}^{(1)}, .., \mathbf{c}^{(m)}\}$ where each $\mathbf{c}^{(j)} \in \{1, .., C\}^b$ represents the choices of chromosomes in the first k levels. The partial score sum is calculated: $\boldsymbol{X}_{\mathbf{c}^{(j)}} = \sum\limits_{i=1}^{b} \boldsymbol{X}_{i\mathbf{c}_i^{(j)}\bullet}$, and dominated partial score vectors are pruned. Then, each of the remaining $\mathbf{c}^{(j)}$'s is expanded into C paths of length $b+1$. A formal step-by-step description

is shown in Algorithm 1. The computational complexity is determined by the number of leafs corresponding to non-dominated vectors considered at each step b, with C^b possible leafs to consider. In the *worst case*, the Branch-and-Bound algorithm enumerates over all leafs, hence it may run in time exponential in M, as shown in the Appendix, Section A.2.

While the worst-case computational complexity of Algorithm 1 is exponential in M, the number of vectors considered may be far lower than C^M in practice. Further pruning can also be achieved by computing upper-and lower-bounds for the optimal loss function as follows: Let $\boldsymbol{X}_\vee(\boldsymbol{X}_\wedge)$ be a vector obtained by summing over all chromosomes i the vector obtained by taking for each coordinate k the maximum (minimum) of X_{ijk} over $j \in [C]$. Then $\mathcal{L}(\boldsymbol{X}_\wedge) \leq \mathcal{L}(\boldsymbol{X}_*) \leq \mathcal{L}(\boldsymbol{X}_\vee)$. Solutions violating this bound are also pruned as part of the algorithm in Step 5.

For simulated problem instances, using the model in eq. (26), the *average* number of leafs at each stage b was far lower than C^b, and grows roughly as $C^{b/2}$, as shown for example in Figure 2(a,b), enabling the usage of Algorithm 1 in practice for large problem instances (e.g. $C = 2, M = 45$).

Algorithm 1 Choosing Best Embryo (Branch-and-Bound)

Input: $\boldsymbol{X}$ - 3^{rd}-order tensor of polygenic scores.
Parameters: $\mathcal{L}$ - a monotone loss function.
1: Initialize the tree of selections $\Gamma \leftarrow \{\mathbf{c}^{(1)}, ..., \mathbf{c}^{(C)}\}$.
2: **for** $b = 2$ to M **do**
3: **for** $j = 1$ to $|\Gamma|$ **do**
4: Compute the partial score vectors $\boldsymbol{X}_{\mathbf{c}^{(j)}} = \sum_{i=1}^{b} \boldsymbol{X}_{i\mathbf{c}_i^{(j)}\bullet}$.
5: Filter all dominated vectors, i.e. set $\Gamma \leftarrow \{c^{(j)} \; s.t. \; \nexists k : \boldsymbol{X}_{\mathbf{c}^{(k)}} \prec \boldsymbol{X}_{\mathbf{c}^{(j)}}\}$.
6: **for** $j = 1$ to $|\Gamma|$ **do**
7: **for** $l = 1$ to C **do**
8: add a new path of length b by expanding $\mathbf{c}^{(j)}$: $\Gamma \leftarrow \Gamma \cup (\mathbf{c}^{(j)}, l)$.
9: Set $\Gamma \leftarrow \Gamma \setminus c^{(j)}$.
10: **for** $j = 1$ to $|\Gamma|$ **do** Compute $\mathcal{L}(\boldsymbol{X}_{\mathbf{c}^{(j)}})$
11: Output the selection path $\mathbf{c}^{(j^*)}$ for $j^* = \underset{j \in [|\Gamma|]}{argmin}\, \mathcal{L}(\boldsymbol{X}_{\mathbf{c}^{(j)}})$.

Divide-and-conquer We can improve the speed of our algorithm, by dividing the M chromosomes into groups, optimizing each of them separately, and then combining the solution in a manner where sub-vectors that cannot be extended to the optimal solution are filtered out. This procedure significantly improve performance, while still guaranteed to yield an optimal solution for monotone losses. Due to its technical details, it is described in Appendix A.3.

4.2 A Relaxation Algorithm

Algorithm 1 (Branch-and-Bound) is inapplicable for non-monotone loss functions. Moreover, even for monotonic losses, the Branch-and-Bound algorithm could be computationally intensive, taking exponential time in the worst case, hence the need for alternative algorithms.

We encode each selection $c_i \in [T]$ using a one-hot vector: $\mathbf{C}_{i\bullet} = (C_{i1}, .., C_{iT})$ with $C_{ic_i} = 1$ and $C_{ij} = 0$, $\forall j \neq c_i$. Next, we relax the requirement that each $C_{ij} \in \{0, 1\}$, and instead just require: $\mathbf{C}_{i\bullet} \in \Delta_T$, where Δ_T denotes the T-dimensional simplex. Concatenating all selection vectors yields a stochastic matrix $\mathbf{C} \in \Delta_T^M \subset \mathbb{R}_{M\times C}$, and the score is given by $\boldsymbol{X}_{\mathbf{c}} = [\boldsymbol{X} \times_2 \mathbf{C}]\mathbf{1}_M$. This leads to the following *relaxed problem*:

Problem 2. **(relaxation):** Given a 3^{rd} order tensor of scores $\boldsymbol{X} \in \mathbb{R}^{M\times C\times T}$, and a loss function $\mathcal{L} : \mathbb{R}^T \to \mathbb{R}$, find a matrix $\mathbf{C}^* \in \Delta_T^M$ minimizing the loss: $\mathbf{C}^* \equiv \underset{\mathbf{C}\in\Delta_T^M}{argmin}\, \mathcal{L}([\boldsymbol{X} \times_2 \mathbf{C}]\mathbf{1}_M)$.

We solve Problem 2 using projected gradient descent, where each row of $\mathbf{C}$ is projected separately onto the simplex Δ_T as described in [4]. Then,closest vertex of the polytope Δ_T^M to $\mathbf{c}^*$ is given as an approximate solution of the original Problem 1. The details are shown in Algorithm 2. When the loss $\mathcal{L}$ is convex, it is possible to establish convergence guarantees for the relaxed Problem 2 (see e.g. [6]), yet the original Problem 1 is computationally hard in general. For smooth losses, it may be possible to get a closed-form solution using Lagrange multipliers, as is demonstrated for the Stabilizing selection loss in Appendix, Section A.5.

Algorithm 2 Choosing Best Embryo (Relaxation)

Input: $\boldsymbol{X}$ - 3^{rd} order tensor of polygenic scores.

Parameters: $\mathcal{L}$ - a monotone loss function, η - gradient step size, δ - convergence tolerance.

1: Initialize the relaxation matrix uniformly $\mathbf{C}^{(0)} = \frac{1}{C}\mathbf{1}_C\mathbf{1}_T^t$.

2: **while** $|\mathcal{L}(\mathbf{C}^{(t+1)}) - \mathcal{L}(\mathbf{C}^{(t)})| > \delta$ **do**

3: Update $\mathbf{C}^{(t+1)} \leftarrow \mathbb{P}_{\Delta_T^M}(\mathbf{C}^{(t)} - \eta\nabla\mathcal{L})$ and compute the score $\mathcal{L}(\mathbf{C}^{(t+1)})$.

4: Round the resulting matrix $\mathbf{C}^{(t+1)}$ to choose the maximal copy for each chromosome. Output the resulting $\mathbf{c}$ defined by: $c_i = \underset{j=1}{\overset{C}{argmax}}\, C_{ij}^{(t+1)}$.

5 Simulation Results

To examine the utility of the two algorithms, we have implemented them as part of an R package called "EmbryoSelectionCalculator", available at https://github.com/orzuk/EmbryoSelectionCalculator (see additional details in Appendix D). We simulated embryo scores from a Matrix Gaussian distribution (see [19]). We mimicked selection of a single sperm cell and a single oocyte, giving us $M = 22 + 23 = 45$ and $C = 2$, i.e. a search space of size $2^{45} \approx 3.5 \times 10^{13}$. We selected for $T = 5$ diseases with equal prevalence of 0.1, and assumed that the polygenic scores explain 20% of the liability for each disease. The relative proportion of variance explained by each chromosome for all traits was according to Table 2. We assumed an heritability of $h^2 = 50\%$ for all diseases

liabilities, and as a consequence define the covariance matrix Σ of the non-score part ϵ to have 0.8 on the diagonal and 0.65 for the off-diagonal elements.

We used the sum of disease probability loss function from eq. with equal weights, and the (minus) probability of being disease-free loss functions (lines 3,4 in Table 1, respectively). The baseline loss under random selection was, as expected $0.1 \times 5 = 0.5$ for the first loss, and was 0.72 for the second, disease-free probability loss, slightly higher than $0.9^5 \approx 0.59$ if diseases were uncorrelated.

We repeated the simulation 100 times, and each time computed the optimal selection strategy using Algorithms 1 and 2. The results are shown in Figure 2(c,d), as a function of the number of available copies for selection C. For the first loss, the outputs obtained by the two algorithms usually coincided, and on average the loss was reduced by 37%. For the second loss, the relaxation algorithm achieved the same solution as the exact Branch-and-Bound algorithm only in 62 out of 100 simulations, and performed worse in the rest 38 simulations, which can be expected for a non-convex loss. The average reduction for this loss was smaller, at 29%. Perhaps surprisingly, the Branch-and-Bound algorithm was faster for both losses, indicating that the trees grown for this model were always kept small. We therefore recommend using the Branch-and-Bound algorithm, and only if the tree size explodes either prune the tree by using a heuristic of keeping only the top paths at each step, or resorting to the relaxation algorithm.

6 Discussion

We have defined and formulated the chromosomal selection problem, and provided two algorithms for solving it. Our Branch-and-Bound algorithm, while exponential in the worst case, can easily be used empirically for the problem of selecting chromosomes from a single sperm cell and a single oocyte for humans, for monotone loss functions. The relaxation algorithm can handle much larger selection problem, yet the performance of the solution obtained by this algorithm may vary. Developing an efficient algorithm with optimality guarantees for major classes of loss functions is an interesting direction for future research.

While the technology for chromosomal selection is not currently available, we believe that our analysis is insightful as it may guide practitioners in the future regarding the potential utility of such technologies. As technologies improve, it may be reformulate the selection problem and adjust the algorithms to adapt to the availability of scores and the constraints on selection imposed by the technology. For example, recent imaging studies of embryos may provide information on their viability and possibly disease risk, without needing to destructively sequence the embryos. If such techniques mature, they can be combined with our computational method to estimate the score of each chromosomal copy and select based on these estimates.

Finally, while current polygenic scores are linear, improved risk predictions may be achieved in the future using nonlinear scores. Formulating the chromosomal selection problem for such nonlinear scores and dealing with the increased combinatorial complexity will posses algorithmic challenges.

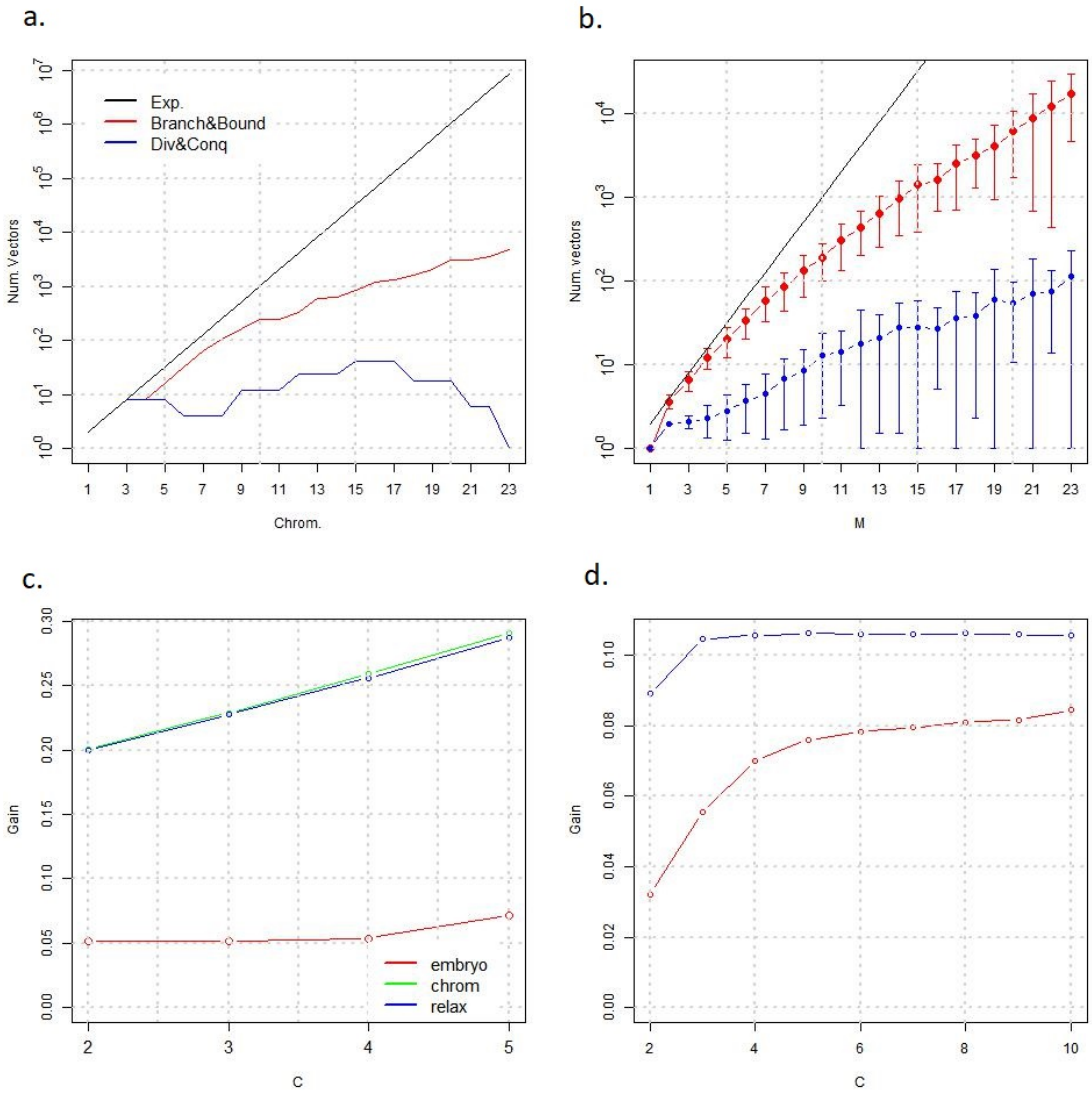

Fig. 2: (a.) The number of vectors enumerated by an algorithm (y-axis) vs. the chromosome considered, in order (x-axis), for a single run in a chromosomal selection problem with $M = 23$ chromosomes, $C = 2$ copies of each, and $T = 5$ traits, with a monotone disease loss. The black, red and blue line correspond respectively to the number of vectors considered in exhaustive search, Algorithm 1 (Branch-and-Bound), and Algorithm 3 (divide and conquer with 8 blocks of 3 or 2 chromosomes each). The number of vectors scanned by Algorithm 3 is approximately four orders of magnitude smaller than exhaustive search, and is maximized at the 15-th chromosome, after which most vectors are excluded due to bound violations. (b.) Average plus/minus one standard deviation of the number of vectors scanned by the different algorithms across 100 simulations, for chromosome selection problems with tensors having different values of M, from 2 to 23. On average, only $\sim$100 vectors are scanned for $M = 23$ for Algorithm 3. (c.) Expected gain for chromosomal selection vs. embryo selection with $M = 45$ chromosomes and $T = 5$ diseases as a function of the number of chromosomal copies C for each chromosome. (c.) For the disease loss (genetic covariance matrices and disease prevalences are the same as in Figure 2), the green curve shows the average gain with optimal selection. The blue curve shows the average gain when using the relaxation Algorithm 2. simulation details are shown in the main text. (d.) For the stabilizing selection non-monotone loss, the relaxation using a closed-form solution is shown to improve upon embryo selection. The gain plateaus at $C = 4$ as the loss approaches zero, vs. an average loss of ≈ 0.2 for random selection.

Bibliography

[1] Bader, B., Kolda, T.: Algorithm 862: Matlab tensor classes for fast algorithm prototyping. ACM Transactions on Mathematical Software (TOMS) **32**(4), 635–653 (2006)

[2] Branwen, G.: Embryo selection for intelligence, https://www.gwern.net/embryo-selection, https://www.gwern.net/Embryo-selection

[3] Bulik-Sullivan, B., Finucane, H., Anttila, V., Gusev, A., Day, F., Loh, P.R., Duncan, L., Perry, J., Patterson, N., Robinson, E., et al.: An atlas of genetic correlations across human diseases and traits. Nature genetics **47**(11), 1236–1241 (2015)

[4] Chen, Y., Ye, X.: Projection onto a simplex. arXiv preprint arXiv:1101.6081 (2011)

[5] Choi, S., Mak, T.S.H., O'Reilly, P.: Tutorial: a guide to performing polygenic risk score analyses. Nature Protocols **15**(9), 2759–2772 (2020)

[6] Correa, R., Lemaréchal, C.: Convergence of some algorithms for convex minimization. Mathematical Programming **62**(1), 261–275 (1993)

[7] Easley IV., C.A., Phillips, B., McGuire, M., Barringer, J., Valli, H., Hermann, B., Simerly, C., Rajkovic, A., Miki, T., Orwig, K., et al.: Direct differentiation of human pluripotent stem cells into haploid spermatogenic cells. Cell reports **2**(3), 440–446 (2012)

[8] Eddelbuettel, D., François, R.: Rcpp: Seamless R and C++ integration. Journal of Statistical Software **40**(8), 1–18 (2011). https://doi.org/10.18637/jss.v040.i08

[9] Falconer, D.: Introduction to Quantitative Genetics. Pearson Education India (1996)

[10] Finucane, H., Bulik-Sullivan, B., Gusev, A., Trynka, G., Reshef, Y., Loh, P.R., Anttila, V., Xu, H., Zang, C., Farh, K., et al.: Partitioning heritability by functional annotation using genome-wide association summary statistics. Nature Genetics **47**(11), 1228 (2015)

[11] Ge, T., Chen, C.Y., Ni, Y., Feng, Y.C.A., Smoller, J.: Polygenic prediction via Bayesian regression and continuous shrinkage priors. Nature Communications **10**(1), 1–10 (2019)

[12] Gould, N., Hribar, M., Nocedal, J.: On the solution of equality constrained quadratic programming problems arising in optimization. SIAM Journal on Scientific Computing **23**(4), 1376–1395 (2001)

[13] Gupta, A., Nagar, D.: Matrix Variate Distributions. Chapman and Hall/CRC (2018)

[14] Halldorsson, B., Eggertsson, H., Moore, K.H., Hauswedell, H., Eiriksson, O., Ulfarsson, M., Palsson, G., Hardarson, M., Oddsson, A., Jensson, B., et al.: The sequences of 150,119 genomes in the UK biobank. Nature pp. 1–9 (2022)

[15] Hansen, T.: Stabilizing selection and the comparative analysis of adaptation. Evolution **51**(5), 1341–1351 (1997)

[16] Hayashi, K., Ogushi, S., Kurimoto, K., Shimamoto, S., Ohta, H., Saitou, M.: Offspring from oocytes derived from in vitro primordial germ cell–like cells in mice. Science **338**(6109), 971–975 (2012)

[17] Hou, Y., Fan, W., Yan, L., Li, R., Lian, Y., Huang, J., Li, J., Xu, L., Tang, F., Xie, X., et al.: Genome analyses of single human oocytes. Cell **155**(7), 1492–1506 (2013)

[18] Jinek, M., Chylinski, K., Fonfara, I., Hauer, M., Doudna, J., Charpentier, E.: A programmable dual-rna–guided dna endonuclease in adaptive bacterial immunity. Science **337**(6096), 816–821 (2012)

[19] Kamath, G.: Bounds on the expectation of the maximum of samples from a gaussian. http://www.gautamkamath.com/writings/gaussian_max.pdf (2015)

[20] Kandel, M., Rubessa, M., He, Y., Schreiber, S., Meyers, S., Matter Naves, L., Sermersheim, M., Sell, G.S., Szewczyk, M., Sobh, N., et al.: Reproductive outcomes predicted by phase imaging with computational specificity of spermatozoon ultrastructure. Proceedings of the National Academy of Sciences **117**(31), 18302–18309 (2020)

[21] Karavani, E., Zuk, O., Zeevi, D., Atzmon, G., Barzilai, N., Stefanis, N., Hatzimanolis, A., Smyrnis, N., Avramopoulos, D., Kruglyak, L., et al.: Screening human embryos for polygenic traits has limited utility. Cell **179**(6), 1424–1435 (2019)

[22] Khera, A., Chaffin, M., Aragam, K., Haas, M., Roselli, C., Choi, S., Natarajan, P., Lander, E., Lubitz, S., Ellinor, P., et al.: Genome-wide polygenic scores for common diseases identify individuals with risk equivalent to monogenic mutations. Nature Genetics **50**(9), 1219–1224 (2018)

[23] Kolda, T., Bader, B.: Tensor decompositions and applications. SIAM Review **51**(3), 455–500 (2009)

[24] Kyrillidis, A., Becker, S., Cevher, V., Koch, C.: Sparse projections onto the simplex. In: International Conference on Machine Learning. pp. 235–243 (2013)

[25] Lambert, S., Gil, L., Jupp, S., Ritchie, S., Xu, Y., Buniello, A., McMahon, A., Abraham, G., Chapman, M., Parkinson, H., et al.: The polygenic score catalog as an open database for reproducibility and systematic evaluation. Nature Genetics **53**(4), 420–425 (2021)

[26] Lencz, T., Backenroth, D., Granot-Hershkovitz, E., Green, A., Gettler, K., Cho, J., Weissbrod, O., Zuk, O., Carmi, S.: Utility of polygenic embryo screening for disease depends on the selection strategy. Elife **10** (2021)

[27] Li, P., Rangapuram, S., Slawski, M.: Methods for sparse and low-rank recovery under simplex constraints. arXiv preprint arXiv:1605.00507 (2016)

[28] McCallum, C., Riordon, J., Wang, Y., Kong, T., You, J., Sanner, S., Lagunov, A., Hannam, T., Jarvi, K., Sinton, D.: Deep learning-based selection of human sperm with high dna integrity. Communications Biology **2**(1), 1–10 (2019)

[29] Pattee, J., Pan, W.: Penalized regression and model selection methods for polygenic scores on summary statistics. PLoS Computational Biology **16**(10), e1008271 (2020)

[30] Paulis, M., Castelli, A., Susani, L., Lizier, M., Lagutina, I., Focarelli, M., Recordati, C., Uva, P., Faggioli, F., Neri, T., et al.: Chromosome transplantation as a novel approach for correcting complex genomic disorders. Oncotarget **6**(34), 35218–35230 (2015)

[31] Paulis, M., Susani, L., Castelli, A., Suzuki, T., Hara, T., Straniero, L., Duga, S., Strina, D., Mantero, S., Caldana, E., et al.: Chromosome transplantation: A possible approach to treat human x-linked disorders. Molecular Therapy-Methods & Clinical Development **17**, 369–377 (2020)

[32] Privé, F., Aschard, H., Carmi, S., Folkersen, L., Hoggart, C., O'Reilly, P., Vilhjálmsson, B.: Portability of 245 polygenic scores when derived from the uk biobank and applied to 9

ancestry groups from the same cohort. The American Journal of Human Genetics **109**(1), 12–23 (2022)

[33] Sanjak, J., Sidorenko, J., Robinson, M., Thornton, K., Visscher, P.: Evidence of directional and stabilizing selection in contemporary humans. Proceedings of the National Academy of Sciences **115**(1), 151–156 (2018)

[34] Schmalhausen, I.: Factors of evolution: the theory of stabilizing selection. (1949)

[35] Treff, N., Eccles, J., Lello, L., Bechor, E., Hsu, J., Plunkett, K., Zimmerman, R., Rana, B., Samoilenko, A., Hsu, S., et al.: Utility and first clinical application of screening embryos for polygenic disease risk reduction. Frontiers in Endocrinology **10**, 845 (2019)

[36] Treff, N., Eccles, J., Marin, D., Messick, E., Lello, L., Gerber, J., Xu, J., Tellier, L.: Preimplantation genetic testing for polygenic disease relative risk reduction: Evaluation of genomic index performance in 11,883 adult sibling pairs. Genes **11**(6), 648 (2020)

[37] Uno, N., Takata, S., Komoto, S., Miyamoto, H., Nakayama, Y., Osaki, M., Mayuzumi, R., Miyazaki, N., Hando, C., Abe, S., et al.: Panel of human cell lines with human/mouse artificial chromosomes. Scientific Reports **12**(1), 1–13 (2022)

[38] Visscher, P.M., Wray, N.R., Zhang, Q., Sklar, P., McCarthy, M.I., Brown, M.A., Yang, J.: 10 years of GWAS discovery: Biology, function, and translation. The American Journal of Human Genetics **101**, 5 (2017)

[39] Visscher, P., Wray, N., Zhang, Q., Sklar, P., McCarthy, M., Brown, M., Yang, J.: 10 years of gwas discovery: biology, function, and translation. The American Journal of Human Genetics **101**(1), 5–22 (2017)

[40] Yang, J., Manolio, T., Pasquale, L., Boerwinkle, E., Caporaso, N., Cunningham, J., De Andrade, M., Feenstra, B., Feingold, E., Hayes, M., et al.: Genome partitioning of genetic variation for complex traits using common snps. Nature Genetics **43**(6), 519 (2011)

[41] Zhu, F., Nair, R., Fisher, E., Cunningham, T.: Humanising the mouse genome piece by piece. Nature Communications **10**(1), 1–13 (2019)

Appendix

A Algorithms and Optimization Details

A.1 Notations

For a matrix X, we denote by $\text{vec}(X)$ the column vector obtained by stacking the columns of X, from first to last. Similarly, for a 3^{rd}-order tensor X, we denote by $\text{mat}(X)$ the matrix obtained by stacking the 2^{nd}-order fibers of X, from first to last.

There are 2^T possible binary vectors of length T, with each such vector $\mathbf{d} \in \{0,1\}^T$, corresponding to an orthant $O_{\mathbf{d}} \subset \mathbb{R}^T$ defined as $O_{\mathbf{d}} \equiv \left\{(x_1,..,x_T)\ s.t.\ (-1)^{d_i}x_i < 0,\ \forall i \in [T]\right\}$.. These 2^T orthants form a disjoint union of $\mathbb{R}^T$ (ignoring equalities with the axes).

In similar to eq. (1), for a vector $\mathbf{V} \in \mathbb{R}^p$, define the *3-mode* tensor-by-vector product as the matrix $[\boldsymbol{X} \times_3 \mathbf{V}] \in \mathbb{R}_{m\times n}$, with elements defined by:

$$[\boldsymbol{X} \times_3 \mathbf{V}]_{ij} \equiv \sum_{k=1}^{p} X_{ijk}V_k,\ \forall i \in [m], \forall j \in [n]. \tag{5}$$

Element-wise notations For two matrices A, B of the same size, their Hadamard product $\odot$ is defined as a matrix obtained by element-wise multiplication of their elements, i.e. $[A \odot B]_{ij} = a_{ij}b_{ij}$. Similarly, we define their entry-wise minimum and maximum $A\bigvee B$ and $A\bigwedge B$ as $[A\bigvee B]_{ij} = \max(a_{ij}, b_{ij})$ and $[A\bigwedge B]_{ij} = \min(a_{ij}, b_{ij})$. For a real number $\alpha \in \mathbb{R}$, the Hadamard power $^{\circledcirc}$ of A is defined by taking raising each element to power α, i.e. $[A^{\circledcirc}]_{ij} = a_{ij}^{\alpha}$.

In the same spirit, the row-wise maximum and minimum vectors are denoted as $A_{\bigvee}, A_{\bigwedge}$, where $[A_{\bigvee}]_i = \max_j a_{ij}$ and $[A_{\bigwedge}]_i = \min_j a_{ij}$. Finally, we can similarly define a vector of indexes obtained by taking the index maximizing/minimizing the elements of A in each row, i.e. $A_{arg\vee}$ is defined as $[A_{arg\vee}]_i = \underset{j}{argmax}\, a_{ij}$ and similarly for $A_{arg\wedge}$.

A.2 Branch-and-Bound

Claim. In the worst case, the number of non-dominated vectors at stage b of Algorithm 1 is C^b.

Proof. We construct the chromosome scores as follows: Draw $u_{ij} \overset{i.i.d.}{\sim} U[0,1]$, $\forall i \in [M]$, $\forall j \in [C]$. For each u_{ij}, set the vector $X_{ij\bullet} \equiv \left(u_{ij}, u_{ij}, ..., u_{ij}, (T-1)u_{ij}\right)$. At stage b of Algorithm 1, any vector present is of the form $\sum_{i=1}^{b} X_{ic_i\bullet} = \left(u, u, ..., u, (T-1)u\right)$ for some $c_i \in [C]$ and for some $u \in \mathbb{R}^+$. With probability one all such u values for different linear combinations are different. Any vector $\left(u, u, ..., u, (T-1)u\right)$ is Pareto-optimal among any set of vectors all sharing the same direction, hence at stage b we get a set of C^b distinct, Pareto-optimal vectors, and in this case the Branch-and-Bound Algorithm does not exclude any of them, reaching at the final stage to all C^M linear combinations.

A.3 Divide-and-conquer

Remark 1. Suppose that we divide the M chromosomes into b blocks, and let $\cup_{i=1}^{n} B_i = [M]$ be a disjoint union of $[M]$. Let $\Gamma^{(i)} \equiv \{\boldsymbol{X}^{(i)}_{\mathbf{c}^{(1)}}, .., \boldsymbol{X}^{(i)}_{\mathbf{c}^{(S_i)}}\}$ be the set of Pareto-optimal vectors obtained by running Algorithm 1 on $\boldsymbol{X}_{B_i \bullet\bullet}$, for $i = 1, .., b$. Furthermore, define $\boldsymbol{X}^{(i)}_{\textcircled{\wedge}}$ ($\boldsymbol{X}^{(i)}_{\textcircled{\vee}}$) as the vectors obtained by taking in coordinate j the maximum (minimum) over all $\boldsymbol{X}^{(i)}_{\mathbf{c}^{(j)}}, \forall j \in [S_i]$. Then:

$$\mathcal{L}(\sum_{i=1}^{b} \boldsymbol{X}^{(i)}_{\textcircled{\vee}}) \leq \mathcal{L}(\boldsymbol{X}_*) \leq \mathcal{L}(\sum_{i=1}^{b} \boldsymbol{X}^{(i)}_{\textcircled{\wedge}}). \tag{6}$$

Based on eq. (6), it is possible to design an algorithm that approximates the true loss by providing upper and lower bounds. When these bounds are close to each other, we may stop the algorithm, while if they are far from each other, we may continue by taking the union of B_i's to get fewer and larger blocks.

We can also exclude some vectors, in similar to above. Namely, Let $\boldsymbol{X}^{(i)}_*$ be the optimal vector for $\boldsymbol{X}_{B_i \bullet\bullet}$. Then:

$$\mathcal{L}(\boldsymbol{X}_*) \leq \mathcal{L}(\sum_{i=1}^{b} \boldsymbol{X}^{(i)}_*). \tag{7}$$

The upper-bound $\mathcal{L}(\sum_{i=1}^{b} \boldsymbol{X}^{(i)}_*)$ is tighter (smaller) than the upper-bound $\mathcal{L}(\sum_{i=1}^{b} \boldsymbol{X}^{(i)}_{\textcircled{\wedge}})$.

We can use the bound to get a divide-and-conquer approach detailed in Algorithm .

Algorithm 3 Choosing Best Embryo (Branch-and-Bound Divide-and-Conquer)

Input: $\boldsymbol{X}$ - 3^{rd}-order tensor of polygenic scores.

Parameters: $\mathcal{L}$ - a monotone loss function, $\{B_1, .., B_b\}$ partition of $[M]$ to sub-blocks.

for $i = 1$ to b **do**

  Run Algorithm 1 on $\boldsymbol{X}_{B_i \bullet\bullet}$ to get the optimal solution $\boldsymbol{X}^{(i)}_*$, and Pareto-optimal set $\Gamma^{(i)}$. Compute also $\boldsymbol{X}^{(i)}_{\textcircled{\wedge}}$ by taking the minimum of each column.

for $i = 1$ to b **do**

  for $j = 1$ to $|\Gamma^{(i)}|$ **do**

    if $\mathcal{L}(\sum_{l=i}^{b} \boldsymbol{X}^{(j)}_*) < \mathcal{L}(\boldsymbol{X}^{(i)}_j + \sum_{l=i+1}^{b} \boldsymbol{X}^{(l)}_{\textcircled{\wedge}})$ **then**

      filter the j-th solution.

    else

      for $l = 1$ to $|\Gamma^{(i+1)}|$ **do**

        add a new path of length $\sum_{i=1}^{b} |B_i|$ by expanding $\mathbf{c}^{(j)}$: $\Gamma^{(i+1)} \leftarrow \Gamma^{(i+1)} \cup (\mathbf{c}^{(j)}, \mathbf{c}^{(l)})$.

for $j = 1$ to $|\Gamma^{(b)}|$ **do**

  Compute $\mathcal{L}(\boldsymbol{X}_{\mathbf{c}^{(j)}})$

Output the selection path $\mathbf{c}^{(j^*)}$ for $j^* = \underset{j \in [|\Gamma^{(b)}|]}{argmin}\, \mathcal{L}(\boldsymbol{X}_{\mathbf{c}^{(j)}})$.

A.4 Computing the Gradient

We show for example the gradient computation for the stabilizing selection loss and for the disease loss. We also show that the relaxed optimization problem 2 is convex for the first case, and not convex for the second case.

1. Consider the stabilizing selection loss: $\mathcal{L}(\boldsymbol{X}_{\mathbf{c}}) = \sum_{i=1}^{T} w_i \boldsymbol{X}_{\mathbf{c}_i}^2$. In terms of the relaxed variables, the loss becomes:

$$\mathcal{L}(\mathbf{C}) = ||\mathbf{w}^{(1/2)} \odot ([\boldsymbol{X} \times_2 \mathbf{C}]\mathbf{1}_C)||^2 \tag{8}$$

where the Hadamard power $^{(1/2)}$ and the Hadamard product $\odot$ are taken element-wise. We next compute the gradient and show that the problem is convex:

Claim. The loss in eq. (8) is convex in $\mathbf{C}$.

Proof. The gradient is given by:

$$\begin{aligned}
\frac{\partial \mathcal{L}(\mathbf{C})}{\partial C_{ij}} &= 2 \sum_{k=1}^{T} w_k X_{ijk} \sum_{m=1}^{M} \sum_{c=1}^{C} X_{mck} C_{mc} \\
&= 2 \sum_{k=1}^{T} w_k X_{ijk} \sum_{m=1}^{M} [\boldsymbol{X} \times_2 \mathbf{C}]_{mk} \\
&= 2 \sum_{k=1}^{T} W_k X_{ijk} \left[\mathbf{1}_M [\boldsymbol{X} \times_2 \mathbf{C}]\right]_k \\
&= 2 \left[\boldsymbol{X} \times_3 \left[\mathbf{w} \odot \left[\mathbf{1}_M [\boldsymbol{X} \times_2 \mathbf{C}]\right]\right]\right]_{ij}
\end{aligned} \tag{9}$$

Therefore, the gradient is:

$$\nabla \mathcal{L}(\mathbf{C}) = 2 \left[\boldsymbol{X} \times_3 \left[\mathbf{w} \odot \left[\mathbf{1}_M [\boldsymbol{X} \times_2 \mathbf{C}]\right]\right]\right]. \tag{10}$$

The Hessian elements are given by:

$$\frac{\partial^2 \mathcal{L}(\mathbf{C})}{\partial C_{ij} \partial C_{mc}} = 2 \sum_{k=1}^{T} w_k X_{ijk} X_{mck}. \tag{11}$$

If we vectorize the matrix $\mathbf{C} \in \mathbb{R}_{M \times C}$ to get a vector $\text{vec}(\mathbf{C}) \in \mathbb{R}^{MC}$, and similarly get $\text{mat}(\boldsymbol{X}) \in \mathbb{R}^{MC \times T}$, the Hessian can be written in matrix form as:

$$\mathbf{H}\big(\mathcal{L}(\mathbf{C})\big) = \left[(2\mathbf{w})^{(1/2)} \mathbf{1}_T \text{mat}(\boldsymbol{X})\right] \left[(2\mathbf{w})^{(1/2)} \mathbf{1}_T \text{mat}(\boldsymbol{X})\right]^t \tag{12}$$

Hence the Hessian is positive semi-definite, therefore the loss is convex in $\mathbf{C}$.

2. Recall the disease loss $\sum_{i=1}^{T} w_i P(D_i = 1|\boldsymbol{X}_{\mathbf{c}})$, with the conditional disease probabilities given by the liability-threshold model, $P(D_i = 1|\boldsymbol{X}_{\mathbf{c}}) = \Phi\Big(\frac{\mathbf{z}_{\mathbf{K}_i} - \boldsymbol{X}_{\mathbf{c}_i}}{\sqrt{{\sigma_{ii}^{(g)}}^2 + {\sigma_{ii}^{(e)}}^2}}\Big)$. Taking the partial derivatives with respect to the relaxation variables yields:

$$\begin{aligned}\frac{\partial \mathcal{L}(\mathbf{C})}{\partial C_{ij}} &= \sum_{k=1}^{T} w_k \frac{\partial \Phi\Big(\frac{\mathbf{z}_{\mathbf{K}_k} - \boldsymbol{X}_{\mathbf{c}_k}}{\sqrt{{\sigma_{kk}^{(g)}}^2 + {\sigma_{kk}^{(e)}}^2}}\Big)}{\partial C_{ij}} \\ &= -\sum_{k=1}^{T} w_k \phi\Big(\frac{\mathbf{z}_{\mathbf{K}_k} - \boldsymbol{X}_{\mathbf{c}_k}}{\sqrt{{\sigma_{kk}^{(g)}}^2 + {\sigma_{kk}^{(e)}}^2}}\Big) \frac{X_{ijk}}{\sqrt{{\sigma_{kk}^{(g)}}^2 + {\sigma_{kk}^{(e)}}^2}}\end{aligned} \tag{13}$$

and the gradient is, in matrix form and using the tensor-by-vector product:

$$\nabla \mathcal{L}(\mathbf{C}) = -\boldsymbol{X} \times_3 \Big[\mathbf{w} \odot diag(\boldsymbol{\Sigma}^{(\boldsymbol{Y})} + \boldsymbol{\Sigma}^{(\boldsymbol{e})})^{-1} \odot \phi\big((\mathbf{z}_{\mathbf{K}} - \boldsymbol{X}_{\mathbf{c}}) diag(\boldsymbol{\Sigma}^{(\boldsymbol{Y})} + \boldsymbol{\Sigma}^{(\boldsymbol{e})})^{-1}\big)\Big]. \tag{14}$$

We next compute the Hessian matrix,

$$\begin{aligned}\frac{\partial^2 \mathcal{L}(\mathbf{C})}{\partial C_{ij} \partial C_{mc}} &= \sum_{k=1}^{T} w_k \frac{\mathbf{z}_{\mathbf{K}_k} - \boldsymbol{X}_{\mathbf{c}_k}}{\sigma_{kk}^{(T,\epsilon)}} \phi\Big(\frac{\mathbf{z}_{\mathbf{K}_k} - \boldsymbol{X}_{\mathbf{c}_k}}{\sigma_{kk}^{(T,\epsilon)}}\Big) \frac{X_{ijk} X_{mck}}{\sigma_{kk}^{(T,\epsilon)}} \\ &\equiv \sum_{k=1}^{T} \alpha_k X_{ijk} X_{mck},\end{aligned} \tag{15}$$

where $\alpha_k \equiv \frac{w_k (\mathbf{z}_{\mathbf{K}k} - \boldsymbol{X}_{\mathbf{c}k})}{{\sigma_{kk}^{(g)}}^2 + {\sigma_{kk}^{(e)}}^2} \phi\Big(\frac{\mathbf{z}_{\mathbf{K}k} - \boldsymbol{X}_{\mathbf{c}k}}{\sqrt{{\sigma_{kk}^{(g)}}^2 + {\sigma_{kk}^{(e)}}^2}}\Big)$. We have $sign(\alpha_k) = sign(\mathbf{z}_{\mathbf{K}_k} - \boldsymbol{X}_{\mathbf{c}_k})$ therefore the sign of α_k changes as we change $\boldsymbol{X}_{\mathbf{c}_k}$, hence the loss is not convex in $\mathbf{C}$.

A.5 A Closed-form Relaxation

For the stabilizing selection loss, it is possible to obtain a closed-form solution to the relaxed Problem 2 of minimizing a quadratic loss under linear constraints by adding Lagrange multipliers [12]. Define:

$$f(\mathbf{C}, \lambda) \equiv \mathcal{L}(\mathbf{C}) + \sum_{j=1}^{M} \lambda_j (1 - \sum_{i=1}^{C} C_{ij}). \tag{16}$$

Then:

$$\frac{\partial f(\mathbf{C}, \lambda)}{\partial C_{ij}} = 2\Big[\boldsymbol{X} \times_3 \Big[\mathbf{w} \odot \big[\mathbf{1}_M [\boldsymbol{X} \times_2 \mathbf{C}]\big]\Big]\Big]_{ij} - \lambda_j. \tag{17}$$

Taking $\frac{\partial f(\mathbf{C}, \lambda)}{\partial C_{ij}} = 0 \ \forall i \in [C], \forall j \in [M]$, we can represent the above in matrix form:

$$\mathbf{1}_C^t [\frac{1}{2} \mathbf{A}^{(i,j)} \odot \mathbf{C}] \mathbf{1}_M = \lambda_i, \ \forall i \in [C], \forall j \in [M] \tag{18}$$

where $\mathbf{A}^{(i,j)} \in \mathbb{R}_{M \times C}$ is defined by: $A_{mc}^{(i,j)} \equiv 2 \sum_{k=1}^{T} w_k X_{ijk} X_{mck}$.

We can stack columns of $\mathbf{C}$, and similarly stack fibers of the fourth-order tensor $\mathbf{A}$, to get the following problem:

$$Minimize \quad \frac{1}{2}\text{vec}(\mathbf{C})^t \text{mat}(\mathbf{A})\text{vec}(\mathbf{C}) \quad s.t. \quad E\,\text{vec}(\mathbf{C}) = \mathbf{1}_M \tag{19}$$

where $\text{mat}(\mathbf{A}) \in \mathbb{R}_{MC\times MC}$ is a matrix in which each row contains the rows of the matrix $\mathbf{A}^{(i,j)}$ concatenated, $\text{vec}(\mathbf{C}) \in \mathbb{R}^{MC}$ is a vector obtained by stacking the columns of $\mathbf{C}$, and $E \in \mathbb{R}_{M\times MC}$ is a matrix encoding the equality constraints $\sum_{j=1}^{C} C_{ij} = 1$ given by $e_{ij} = 1\ \forall i \in [M], \forall j \in [C(i-1)+1, Ci]$ and $e_{ij} = 0$ otherwise.

The solution for the above problem is given via Lagrange multipliers as a solution of the linear system:

$$\begin{pmatrix} \text{mat}(\mathbf{A}) & E^t \\ E & \mathbf{0}_{M\times M} \end{pmatrix} \begin{pmatrix} \text{vec}(\mathbf{C}) \\ \lambda \end{pmatrix} = \begin{pmatrix} \mathbf{0}_{MC} \\ \mathbf{1}_M \end{pmatrix} \tag{20}$$

Since $\text{mat}(\mathbf{A})$ is a sum of T matrices of rank 1, we have $rank\big(\text{mat}(\mathbf{A})\big) \leq T$ and

$$rank\left(\begin{pmatrix} \text{mat}(\mathbf{A}) & E^t \\ E & \mathbf{0}_{M\times M} \end{pmatrix}\right) \leq T + 2M. \tag{21}$$

When $T + 2M < (C+1)M$ the above system has an infinite subspace of solutions. When $T + 2M \geq (C+1)M$, there is typically a unique solution for $\mathbf{C}$ obtained by solving the above system, giving us:

$$\begin{pmatrix} \text{vec}(\mathbf{C}) \\ \lambda \end{pmatrix} = \begin{pmatrix} \text{mat}(\mathbf{A}) & E^t \\ E & \mathbf{0}_{M\times M} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{0}_{MC} \\ \mathbf{1}_M \end{pmatrix} \tag{22}$$

and the solution vector $\mathbf{c}$ is obtained by:

$$\mathbf{c} = \Big[mat\big(\text{vec}(\mathbf{C})\big)\Big]_{\overbrace{arg\vee}} \tag{23}$$

where the outer mat operation is reshaping the solution vector $\text{vec}(\mathbf{C})$ into M consecutive equally-sized vectors and stacking them together to form a matrix, in which each row is maximized.

The closed-form solution in eqs. (22,23) can be used instead of Algorithm 2 for the stabilizing selection loss.

When $T + 2M < (C+1)M$ the inverse in eq. (22) can be replaced by the Moore-Penrose pseudo-inverse, yielding the minimal Euclidean-norm solution $\text{vec}(\mathbf{C})$ which is rounded to get $\mathbf{c}$.

Regularization The relaxation in the previous section may yield outputs with many non-zero entries, that are far away from the vertices of the polytope of stochastic matrices. To obtain a solution closer to one of the vertices we add a sparsity-promoting term to the optimization problem 2. The standard L_1 regularization often employed to promote sparsity is inappropriate here, since for every row we have already $\sum_{j=1}^{C} c_{ij} = 1$ hence the elements sum is constant. Several

previous works have suggested algorithms for sparse projection and optimization over the simplex [24, 27]. Our space is a Cartesian product of multiple simplices, where each simplex representing a different row of the matrix $\mathbf{C}$, and we employ a similar technique to promote sparsity. Specifically, we add to the optimization criteria in eq. (2) a negative quadratic loss term [27]: $-\eta||\mathbf{C}||_F^2$, where $\eta < 0$ and $||\cdot||_F^2$ is the squared Frobenius norm. This term promotes solutions with a high Frobenius norm, that are likely to be concentrated on a few entries. Incorporating the additional term in the optimization is straightforward. For example, the term $-2\eta\mathbf{C}$ is added gradient of the loss in Algorithm 2. The closed-form solution for the regularized problem with the stabilizing selection loss is obtained by simply replacing the term $\text{mat}(\mathbf{A})$ with $\text{mat}(\mathbf{A}) - \eta I_{MC}$ in eqs. (19-22). In similar to Ridge regression, the addition of the regularization term yields a unique solution even when the matrix in the left hand side of eq. (21) is singular and the least square solution is not unique, as is the case whenever $T + 2M < (C + 1)M$.

B Quantitative Genetics

To put the abstract problem presented in the previous section in the context of current practice in embryo selection, we describe here a simple quantitative genetics model for embryo selection for multiple quantitative traits and diseases.

B.1 A Statistical Model for Chromosomal Selection

We describe here a statistical model for the joint distribution of the scores and non-score components determining a phenotype in a set of C genomes and for T complex traits. The model is related to and extends models used in [21, 26] for multiple traits, with two main differences: First, we model explicitly the joint distribution of the individual chromosomes' scores, wheres in [21, 26] a model was given for the entire genomic score. Second, [21, 26] considered embryos derived from the same two parents, yielding a specific genetic relationship matrix $\Sigma^{(C)}$ representing the Identity-by-Descent sharing of siblings, while in our case the genetic relationship matrix may be more general depending on the selection scenario.

We assume that the genetic architecture of the traits is infinitesimal, namely that there are numerous causal variants, uniformly distributed along the genome. Denote the matrix of quantitative trait values as $\mathbf{Z} \in \mathbb{R}_{C\times T}$, where Z_{ij} denotes the value of trait j for the i-th copy. We can decompose $\mathbf{Z}$ as follows:

$$\mathbf{Z} = (\boldsymbol{X} + \boldsymbol{Y}) \times_3 \mathbf{1}_M + \boldsymbol{\epsilon} \tag{24}$$

where the error term $\boldsymbol{Y}$ represents a tensor of genetic components not accounted for by the scores $\boldsymbol{X}$, and the error term $\boldsymbol{\epsilon}$ represents a matrix of environmental components, both having zero mean.

We assume that all the traits have mean zero and variance 1, and further that the individual chromosome scores also have zero mean.

We further assume that the distribution of the polygenic scores $\boldsymbol{X}$ is approximately Normal in each embryo (due to the polygenic nature of most complex traits [38]), and that the joint distribution of the polygenic scores over n embryos is multivariate Gaussian,

Consider T traits normalized to have zero mean and unit variance. Let $\boldsymbol{X}^{(e)} \equiv [\boldsymbol{X} \times_3 \mathbf{1}_M] \in \mathbb{R}_{C\times T}$ be a matrix of polygenic scores for the C copies, obtained by summing the individual chromosome scores $\boldsymbol{X}_{i\bullet\bullet}$, and similarly let $\boldsymbol{Y}^{(e)} \equiv [\boldsymbol{Y} \times_3 \mathbf{1}_M] \in \mathbb{R}_{C\times T}$. The vector of polygenic score for a single genomic copy for all traits $\boldsymbol{X}^{(e)}_{i\bullet}$ has a covariance matrix $\boldsymbol{\Sigma^{(X)}}$ under a Normal model:

$$\boldsymbol{X}^{(e)}_{i\bullet} \sim MVN(\mathbf{0_T}, \boldsymbol{\Sigma^{(X)}}) \tag{25}$$

where $diag(\boldsymbol{\Sigma^{(X)}}) = (\sigma^{(X)}_{11}, .., \sigma^{(X)}_{TT})$ contains the variance explained by the polygenic scores of each trait, and the off-diagonal elements of $\boldsymbol{\Sigma^{(X)}}$ represent pleiotropic effects. For C full-genome copies we obtain a $C \times T$ matrix of polygenic scores with a matrix Normal distribution [13]:

$$\boldsymbol{X}^{(e)} \sim MN_{C\times T}(\mathbf{0_{C\times T}}, \boldsymbol{\Sigma^{(C)}}, \boldsymbol{\Sigma^{(X)}}). \tag{26}$$

The matrix $\boldsymbol{\Sigma^{(C)}}$ represents (twice) the kinship coefficients between the C full-genome copies. For example, when the copies represent sibling embryos (as is the case for embryo selection), $\boldsymbol{\Sigma^{(C)}} = \frac{1}{2}[I_C + \mathbf{1}_C\mathbf{1}^t_C]$. We assume that the chromosome scores are independent, with the scores matrix of each chromosome having the matrix Normal distribution:

$$\boldsymbol{X}_{i\bullet\bullet} \sim MN_{C\times T}(\mathbf{0_{C\times T}}, \alpha_i^2\boldsymbol{\Sigma^{(C)}}, \boldsymbol{\Sigma^{(X)}}), \tag{27}$$

where α_i^2 is the proportion of genetic variance explained by chromosome i. The genetic variances satisfy $\sum i = 1^M \alpha_i^2 = 1$, and this proportion is assumed to be the same for all traits, a consequence of the infinitesimal model and provided that the relative density of causal variants across the genome is similar across traits.

This contributions determines the utility of chromosomal selection, and are expected to be roughly proportional to chromosomes' length or to their number of genes. Here, we show a numerical analysis with α_i^2 the (normalized) chromosomes lengths as in [2], shown in Table 2. The actual coefficients α_i^2 may deviate from this rough estimate and from trait to trait, based on the distribution of causal alleles along the genome for each trait. Methods for partitioning heritability [10, 40] can be used to estimate these coefficients for specific traits, and in case of significant deviations, Eq. (27) can be modified accordingly.

Chrom.	1	2	3	4	5	6	7	8	9	10	11	12
Length	0.082	0.080	0.066	0.063	0.060	0.056	0.053	0.048	0.046	0.044	0.045	0.044
Chrom.	13	14	15	16	17	18	19	20	21	22	X	Y
Length	0.0378	0.035	0.034	0.030	0.028	0.027	0.019	0.021	0.015	0.017	0.052	0.0

Table 2: Relative human chromosomes lengths

Similarly, the non-score genetic components are modeled as:

$$\begin{aligned} \boldsymbol{Y}^{(e)} &\sim MN_{C\times T}(\mathbf{0}_{\boldsymbol{C}\times\boldsymbol{T}}, \boldsymbol{\Sigma}^{(\boldsymbol{C})}, \boldsymbol{\Sigma}^{(\boldsymbol{Y})}) \\ \boldsymbol{\epsilon} &\sim MN_{C\times T}(\mathbf{0}_{\boldsymbol{C}\times\boldsymbol{T}}, \boldsymbol{I}_{\boldsymbol{C}}, \boldsymbol{\Sigma}^{(\boldsymbol{e})}) \end{aligned} \tag{28}$$

where $\boldsymbol{\Sigma}^{(\boldsymbol{X})}+\boldsymbol{\Sigma}^{(\boldsymbol{Y})}+\boldsymbol{\Sigma}^{(\boldsymbol{e})} = \boldsymbol{I}_{\boldsymbol{T}}$. The matrix $\boldsymbol{\Sigma}^{(\boldsymbol{X})}+\boldsymbol{\Sigma}^{(\boldsymbol{Y})}$ is known as the genetic covariance matrix, and can be estimated from GWAS data using e.g. methods like LD-Score-Regression [3]. The diagonal elements $h_i^2 = \sigma_{ii}^{(X)} + \sigma_{ii}^{(Y)}$ are the narrow-sense heritabilities of the traits.

The matrix $\boldsymbol{\Sigma}^{(\boldsymbol{Y})} + \boldsymbol{\Sigma}^{(\boldsymbol{e})}$ is the covariance matrix of the residuals, and determines the conditional distribution of the phenotypes vector conditioned on the scores vector. For simplicity, our model makes several standard assumptions: no shared environment (hence the identity $\boldsymbol{I}_{\boldsymbol{C}}$ is used as a covariance matrix for $\boldsymbol{\epsilon}$), and no assortative mating. If these assumptions are violated, this can be encoded by the covariance matrices of our model.

The expected gain The gain $G_e = G_e(\boldsymbol{X}; \mathcal{L})$ defined in eq. (2) is a random variable, with a sample space over all theoretical sets of C copies. In the following, we will derive the approximate mean of the gain for linear loss functions $\mathcal{L}$, as a function of the loss parameters, and of C, $\boldsymbol{\Sigma}^{(\boldsymbol{C})}$, and $\boldsymbol{\Sigma}^{(\boldsymbol{X})}$.

For embryo selection with a linear loss, selection is performed on the vector of scores $\mathcal{L} = \mathbf{w}^t\boldsymbol{X}^{(e)}$, with the joint distribution:

$$\mathbf{w}^t\boldsymbol{X}^{(e)} \sim MVN(\mathbf{0}_{\boldsymbol{C}}, \mathbf{w}^t\boldsymbol{\Sigma}^{(\boldsymbol{X})}\mathbf{w}\boldsymbol{\Sigma}^{(\boldsymbol{C})}). \tag{29}$$

It was shown in [21] that $\mathbb{E}_{\boldsymbol{X}}G_e = \mathbb{E}_{\boldsymbol{X}}\max(\mathbf{w}^t\boldsymbol{X}_{\bullet 1\bullet}\mathbf{1}_M, \ldots, \mathbf{w}^t\boldsymbol{X}_{\bullet C\bullet}\mathbf{1}_M)$. Moreover,

$$\mathbf{w}^t\boldsymbol{X}_{\bullet j\bullet}\mathbf{1}_M \sim N(0, \mathbf{w}^t\boldsymbol{\Sigma}^{(\boldsymbol{X})}\mathbf{w}),\ \forall j \in [C] \tag{30}$$

and

$$\mathbf{w}^t\boldsymbol{X}^{(e)} \sim MVN(\mathbf{0}_{\boldsymbol{C}}, \mathbf{w}^t\boldsymbol{\Sigma}^{(\boldsymbol{X})}\mathbf{w}\boldsymbol{\Sigma}^{(\boldsymbol{C})}). \tag{31}$$

Using extreme value theory for the above, we get as in [21] the approximate gain from embryo selection:

$$\mathbb{E}_{\boldsymbol{X}}G_e \equiv \mathbb{E}_{\boldsymbol{X}}[\max_{j=1}^{C}(\mathbf{w}^t\boldsymbol{X}_{j\bullet}^{(e)})] \approx 0.77\sqrt{\mathbf{w}^t\boldsymbol{\Sigma}^{(\boldsymbol{X})}\mathbf{w}}\sqrt{\log C}. \tag{32}$$

Next, we will compare this result to the gain obtained from chromosomal selection. For each individual chromosome i, the distribution of the scores vectors is

$$\mathbf{w}^t\boldsymbol{X}_{i\bullet\bullet} \sim MVN(\mathbf{0}_{\boldsymbol{C}}, \alpha_i^2\mathbf{w}^t\boldsymbol{\Sigma}^{(\boldsymbol{X})}\mathbf{w}\boldsymbol{\Sigma}^{(\boldsymbol{C})}). \tag{33}$$

Since selection is performed for each block separately, and using again the asymptotic approximation from [21] for the covariance matrices of individual chromosome's scores, the gain can be

written as:

$$\begin{aligned}\mathbb{E}_{\boldsymbol{X}} G &\equiv \sum_{i=1}^{M} \mathbb{E}_{\boldsymbol{X}} \max_{j=1}^{C}(\mathbf{w}^t \boldsymbol{X}_{ij\bullet}) \\ &\approx \sum_{i=1}^{M} 0.77\sqrt{\alpha_i^2 \mathbf{w}^t \boldsymbol{\Sigma}^{(\boldsymbol{X})}\mathbf{w}}\sqrt{\log C} \\ &\approx \Big(\sum_{i=1}^{M} \alpha_i\Big)\mathbb{E}_{\boldsymbol{X}} G_e. \end{aligned} \tag{34}$$

Hence the expected gain due to chromosomal selection is roughly $\sum_{i=1}^{M} \alpha_i$-fold higher compared to the expected gain from embryo selection in eq. (32). For the α_i values in Table 2, this gives a 4.68-fold difference between the gains.

B.2 Disease Traits

Consider a disease with population prevalence K and let X be the polygenic score with variance explained h^2_{ps} on the liability scale, using the liability threshold model:

$$D = 1_{\{z<\Phi^{-1}(K)\}} \tag{35}$$

with $z = X + \epsilon$. The polygenic scores $X^1, ..., X^n$ can be thought of as liabilities, where the actual disease score modeled as $z_i = X^i + \epsilon_i$ with $\epsilon_i \sim N(0, 1 - h^2_{\text{ps}})$ being random variables representing both the environmental contribution as well as unaccounted for genetic effects. The resulting disease status of each individual is given by thresholding the z_i's, $D_i = 1_{\{z_i<\Phi^{-1}(K)\}}$.

We select the embryo with maximal score $X_{\max}$ as in the quantitative trait example, and denote by i_{max} the index of this embryo. As shown in [26], the risk for disease for the embryo with maximal polygenic score is given by a convolution:

$$\begin{aligned} P(D_{i_{max}} = 1) &= P\left(X_{\max} + \epsilon_{i_{max}} < \Phi^{-1}(K)\right) \\ &= \int_{-\infty}^{\infty} \phi(t)\Phi\left(\frac{\Phi^{-1}(K) - t\sqrt{1 - h^2_{\text{ps}}/2}}{h_{\text{ps}}/\sqrt{2}}\right)^n dt \end{aligned} \tag{36}$$

and the expected (absolute) gain for the single disease loss is:

$$\mathbb{E}_{\boldsymbol{X}} G_e = K - \int_{-\infty}^{\infty} \phi(t)\Phi\left(\frac{\Phi^{-1}(K) - t\sqrt{1 - h^2_{\text{ps}}/2}}{h_{\text{ps}}/\sqrt{2}}\right)^n dt. \tag{37}$$

Multiple Diseases We consider screening for multiple T diseases, with the polygenic risk scores given in eq. (26), and with prevalences vector $\mathbf{K} = (K_1, .., K_T)$. We need to define a loss function, representing the trade-offs of reducing risk for multiple diseases - for example the probability of being disease free. The next Section formalizes the goal of selection for multiple quantitative traits or diseases, and in addition presents the problem of chromosomal selection.

We next define the associated disease status and disease probabilities for chromosomal selection.

Definition 3. *For a vector of prevalences* $\mathbf{K} \in [0,1]^T$ *and a residual covariance matrix* $\boldsymbol{\Sigma}^{(\boldsymbol{Y})} + \boldsymbol{\Sigma}^{(\boldsymbol{e})}$*, let* $\boldsymbol{Y} \times \mathbf{1}_M + \boldsymbol{\epsilon} \sim N(0, \boldsymbol{\Sigma}^{(\boldsymbol{Y})} + \boldsymbol{\Sigma}^{(\boldsymbol{e})})$*. Then, the disease status is a vector random variable* $\mathbf{D} \in \mathbb{R}^T$ *defined as:*

$$\mathbf{D} \equiv \mathbb{1}_{\{\boldsymbol{\epsilon}+\boldsymbol{X}_\mathbf{c}<\mathbf{z}_\mathbf{K}\}} \tag{38}$$

where the indicator function is taken element-wise.

The associated disease probability for a given binary vector $\mathbf{d} \in \{0,1\}^T$ *is*

$$P(\mathbf{D}=\mathbf{d}|\boldsymbol{X}_\mathbf{c}) = \Phi(O_\mathbf{d}; \mathbf{z}_\mathbf{K} - \boldsymbol{X}_\mathbf{c}, \boldsymbol{\Sigma}^{(\boldsymbol{Y})} + \boldsymbol{\Sigma}^{(\boldsymbol{e})}). \tag{39}$$

The marginal disease probability for disease i *and status* $j = 0, 1$ *is given by:*

$$\begin{aligned} P(D_i = j|\boldsymbol{X}_\mathbf{c}) &= \sum_{\mathbf{d}\in\{0,1\}^T} \mathbb{1}_{\{d_i=j\}} P(\mathbf{D}=\mathbf{d}|\boldsymbol{X}_\mathbf{c}) \\ &= \Phi\Big((-1)^j \frac{\boldsymbol{X}_{\mathbf{c}i} - \mathbf{z}_{\mathbf{K}i}}{\sqrt{{\sigma_{ii}^{(g)}}^2 + {\sigma_{ii}^{(e)}}^2}}\Big) \\ &= \Phi\Big((-1)^j \frac{\boldsymbol{X}_{\mathbf{c}i} - \mathbf{z}_{\mathbf{K}i}}{\sqrt{1 - {\sigma_{ii}^{(g)}}^2 - {\sigma_{ii}^{(e)}}^2}}\Big). \end{aligned} \tag{40}$$

Definition 4. *A chromosomal selection loss function* $\mathcal{L}$ *is called a disease-loss function if there are vectors* $\mathbf{w} \in \mathbb{R}^{2^T}, \mathbf{K} \in \mathbb{R}^T$ *and a positive semi definite matrix* $\Sigma^{(T,\epsilon)} \in \mathbb{R}_{T\times T}$ *such that* $\mathcal{L}$ *can be written as follows:*

$$\mathcal{L}(\boldsymbol{X}_\mathbf{c}) = \sum_{\mathbf{d}\in\{0,1\}^T} w_\mathbf{d} P(\mathbf{D}=\mathbf{d}|\boldsymbol{X}_\mathbf{c}) = \sum_{\mathbf{d}\in\{0,1\}^T} w_\mathbf{d} \Phi(O_\mathbf{d}; \mathbf{z}_\mathbf{K} - \boldsymbol{X}_\mathbf{c}, \Sigma^{(T,\epsilon)}). \tag{41}$$

If the loss above can be written as follows for a vector $\mathbf{w} \in \mathbb{R}^T$*:*

$$\mathcal{L}(\boldsymbol{X}_\mathbf{c}) = \sum_{i=1}^{T} w_i P(D_i = 1|\boldsymbol{X}_\mathbf{c}) \tag{42}$$

then the loss function is called a linear disease-loss function.

C Modified Selection Problems

We describe here a few additional scenarios that yield the chromosomal selection Problem 1 or variants of it.

C.1 Gamete Selection

Consider an intermediate case of gamete selection, where it is possible to select a sperm and an oocyte separately, as discussed in [2]. We consider T continuous phenotypes as in Section B.1. Consider C_p haploid sperm cells, and C_m haploid oocyte cells. Denote their scores matrices by $X^{(e),p} \in \mathbb{R}_{C_p \times T}, X^{(e),m} \in \mathbb{R}_{C_m \times T}$

The Gamete selection problem is to select a single sperm cell $i_p \in [C_p]$ and a single oocyte $i_m \in [C_m]$ minimizing the loss $\mathcal{L}(X^{(e),p}_{i_p \bullet} + X^{(e),m}_{i_m \bullet})$. See an illustration in Figure 3(a).

We can compute the expected gain for gamete selection with a linear loss $\mathcal{L} = \mathbf{w}^t(X^{(e),p}_{i_p \bullet} + X^{(e),m}_{i_m \bullet})$ in similar to the derivations for embryo selection. First, in similar to eq. (26), we have

$$\begin{aligned} X^{(e),p} &\sim MN_{C_p \times T}(\mathbf{0}_{\boldsymbol{C_p \times T}}, \boldsymbol{\Sigma^{(C_p)}}, \frac{1}{2}\boldsymbol{\Sigma^{(X)}}). \\ X^{(e),m} &\sim MN_{C_m \times T}(\mathbf{0}_{\boldsymbol{C_m \times T}}, \boldsymbol{\Sigma^{(C_m)}}, \frac{1}{2}\boldsymbol{\Sigma^{(X)}}). \end{aligned} \tag{43}$$

where $\boldsymbol{\Sigma^{(C_p)}}, \boldsymbol{\Sigma^{(C_m)}}$ are the covariance matrices for the sperm and oocyte cells, respectively, with (twice) kinship coefficient of $\frac{1}{2}$, and assuming that the covariance of the scores of any sperm and oocyte cell is zero. We also assume that the trait's variance matrices $\Sigma^{(X)}$ are equal due to symmetry of the maternal and paternal contribution to traits (we ignore here the contribution of the sex chromosomes).

With these matrices, we get for all $i_p \in [C_p]$, $i_m \in [C_m]$:

$$\mathbf{w}^t X^{(e),p}_{i_p \bullet}, \mathbf{w}^t X^{(e),m}_{i_m \bullet} \overset{i.i.d.}{\sim} N(0, \frac{1}{2}\mathbf{w}^t \boldsymbol{\Sigma^{(X)}} \mathbf{w}) \tag{44}$$

and

$$\begin{aligned} \mathbf{w}^t X^{(e),p} &\sim MVN(0, \frac{1}{2}\mathbf{w}^t \boldsymbol{\Sigma^{(X)}} \mathbf{w} \boldsymbol{\Sigma^{(C_p)}}), \\ \mathbf{w}^t X^{(e),m} &\sim MVN(0, \frac{1}{2}\mathbf{w}^t \boldsymbol{\Sigma^{(X)}} \mathbf{w} \boldsymbol{\Sigma^{(C_m)}}). \end{aligned} \tag{45}$$

Following the derivation for a single trait, we get:

$$\begin{aligned} \mathbb{E}_{\boldsymbol{X}} G &\equiv \mathbb{E}_{\boldsymbol{X}}[\max(X^1_{\mathbf{w},p}, .., X^{n_p}_{\mathbf{w},p}) + \max(X^1_{\mathbf{w},m}, .., X^{n_m}_{\mathbf{w},m})] \\ &\approx 0.77\sqrt{\frac{1}{2}\mathbf{w}^t \Sigma^{(X)} \mathbf{w}}[\sqrt{\log C_p} + \sqrt{\log C_m}] \\ &= 0.55\sqrt{\mathbf{w}^t \Sigma^{(X)} \mathbf{w}}[\sqrt{\log C_p} + \sqrt{\log C_m}]. \end{aligned} \tag{46}$$

For example, suppose that we have an equal number of sperm and oocyte cells $C_p = C_m = C$. Then the gain is $\approx 1.1\sqrt{\mathbf{w}^t \Sigma^{(X)} \ \log C}$, a $\sqrt{2}$-factor improvement over the gain from embryo selection with the same C shown in eq. (32).

C.2 Chromosomal Selection for Multiple Sperm Cells and Oocytes

Suppose that we face the scenario in the previous sub-section, except that it is possible to select different chromosomes from different sperm cells, and similarly different chromosomes

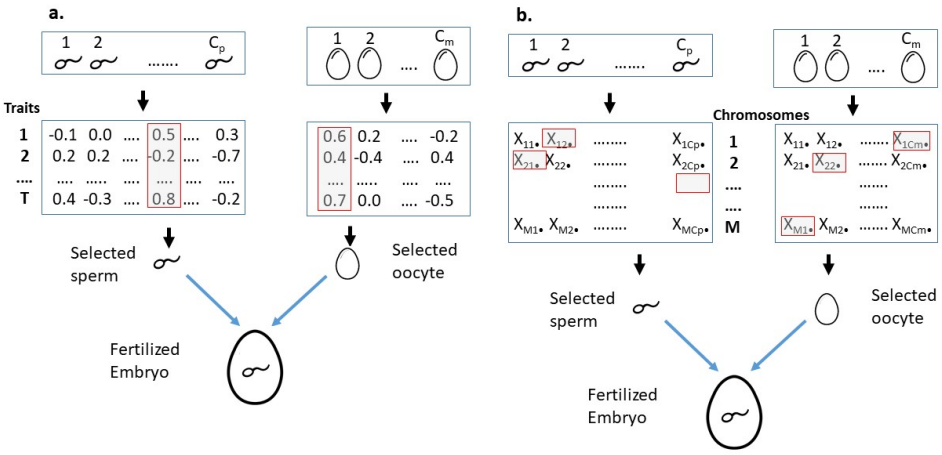

Fig. 3: Illustration of different types of selection based of polygenic scores for T traits. (a.) Gamete selection may enable to choose sperm and oocyte cells based on their polygenic scores or proxy phenotypes, and fertilize the selected oocyte and sperm. (b.) In chromosomal selection, we may select for each chromosome i from a different sperm cell or oocyte j based on the score vector $\boldsymbol{X}_{ij\bullet}$. When the numbers of sperm and oocyte cells are equal, $C_p = C_m = C$, the problem reduces to Problem 1.

from different oocytes (assuming the scores can be computed form the cells in a non-destructive manner). Then, we face a chromosomal selection problem similar to Problem 1, except that the number of available copies may be different for different chromosomes (either C_p or C_m), as shown in Figure 3(b). When, $C_p = C_m$, the problem reduces back to Problem 1.

C.3 Chromosomal Selection from Multiple Diploid Cells

Consider $C/2$ diploid cells (for even C), and suppose that we select for each chromosome two copies in an arbitrary manner for the fertilized embryo (for example, it may be possible to select both copies of the same diploid cells). Then, we face a chromosomal selection problem with $M = 23$, except that two, rather than one copy, is selected from the scores tensor $\boldsymbol{X}$. Algorithms 1 and 2 can be adapted in a rather straightforward manner to handle this case, and their implementation and study remain for future work.

D Implementation Details

The entire algorithms and the simulation study are implemented as part of an R package called "EmbryoSelectionCalculator", available at https://github.com/orzuk/EmbryoSelectionCalculator,

with the functions related to chromosomal selection located in the chrom sub-directory. To speed-up computations, code for finding Pareto-optimal vectors was implemented in cpp, and linked using rcpp [8]. To avoid combinatorial explosion of the Branch-and-Bound algorithm, a heuristic of passing only the top B vectors at each step when the number of partial Pareto-optimal vectors exceeds B was implemented as optional, and used with the default value of $B = 10,000$. An additional optional improvement to the relaxation Algorithm 2 is also implemented: instead of rounding the solution of the relaxed problem $\mathbf{C}^{(t+1)}$ to the nearest vertex of the polytope of stochastic matrices, it is possible to draw at random the selection variables c_i from a categorical distribution with values $\{1, .., C\}$ and with probabilities given by the i-th row of $\mathbf{C}^{(t+1)}$. One can draw independently multiple such vectors (default value: $R = 1,000$) and output the solution minimizing the loss $\mathcal{L}$ among the resulting $\boldsymbol{X}_{\mathbf{c}}$ scores vectors. Additional details about the software implementation and usage are available in the package documentation.