from each voters ranking. If the dropped candidate was number one on some voters list, then the number two candidate becomes that voter's number one choice. The process of counting the number one rankings is then repeated.

We can convert the Hare voting system into a ranking method in the following way. Whichever candidate is dropped first is put in last place, whichever is dropped second is put in second-to-last place, and so on, until the system selects a winner, which is put in first place. The candidates remaining, if any, are placed between the first-place candidate and the candidates who were dropped, in an order determined by running this procedure recursively on just those remaining candidates.

As with Borda Count, the Hare system also fails to satisfy independence of irrelevant alternatives. Consider the following situation in which there are 21 voters that fall into four categories. Voters within a category rank individuals in the same order.

Category	Number of voters in category	Preference order
1	7	abcd
2	6	bacd
3	5	cbad
4	3	dcba

The Hare system would first eliminate d since d gets only three rank one votes. Then it would eliminate b since b gets only six rank one votes whereas a gets seven and c gets eight. At this point a is declared the winner since a has thirteen votes to c's eight votes. So, the final ranking is *acbd*.

Now assume that Category 4 voters who prefer b to a move b up to first place. This keeps their order of a and b unchanged, but it reverses the global order of a and b. In particular, d is first eliminated since it gets no rank one votes. Then c with five votes is eliminated. Finally, b is declared the winner with 14 votes, so the final ranking is *bacd*.

Interestingly, Category 4 voters who dislike a and have ranked a last could prevent a from winning by moving a up to first. Ironically this results in eliminating d, then c, with five votes and declaring b the winner with 11 votes. Note that by moving a up, category 4 voters were able to deny a the election and get b to win, whom they prefer over a.

10.2 Compressed Sensing and Sparse Vectors

Define a *signal* to be a vector $\mathbf{x}$ of length d, and define a *measurement* of $\mathbf{x}$ to be a dot-product of $\mathbf{x}$ with some known vector $\mathbf{a_i}$. If we wish to uniquely reconstruct $\mathbf{x}$ without any assumptions, then d linearly-independent measurements are necessary and sufficient.

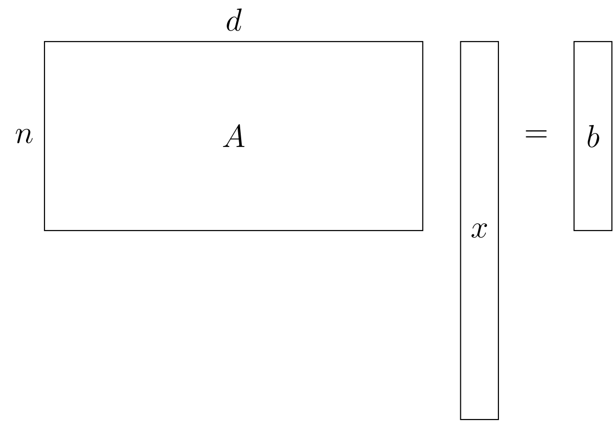

Figure 10.2: $A\mathbf{x} = \mathbf{b}$ has a vector space of solutions but possibly only one sparse solution. If the columns of A are unit length vectors that are pairwise nearly orthogonal, then the system has a unique sparse solution.

Given $\mathbf{b} = A\mathbf{x}$ where A is known and invertible, we can reconstruct $\mathbf{x}$ as $\mathbf{x} = A^{-1}\mathbf{b}$. In the case where there are fewer than d independent measurements and the rank of A is less than d, there will be multiple solutions. However, if we knew that $\mathbf{x}$ is sparse with $s \ll d$ nonzero elements, then we might be able to reconstruct $\mathbf{x}$ with far fewer measurements using a matrix A with $n \ll d$ rows. See Figure 10.2. In particular, it turns out that a matrix A whose columns are nearly orthogonal, such as a matrix of random Gaussian entries, will be especially well-suited to this task. This is the idea of compressed sensing. Note that we cannot make the columns of A be completely orthogonal since A has more columns than rows.

Compressed sensing has found many applications, including reducing the number of sensors needed in photography, using the fact that images tend to be sparse in the wavelet domain, and in speeding up magnetic resonance imaging in medicine.

10.2.1 Unique Reconstruction of a Sparse Vector

A vector is said to be s-sparse if it has at most s nonzero elements. Let $\mathbf{x}$ be a d-dimensional, s-sparse vector with $s \ll d$. Consider solving $A\mathbf{x} = \mathbf{b}$ for $\mathbf{x}$ where A is an $n \times d$ matrix with $n < d$. The set of solutions to $A\mathbf{x} = \mathbf{b}$ is a subspace. However, if we restrict ourselves to sparse solutions, under certain conditions on A there is a unique s-sparse solution. Suppose that there were two s-sparse solutions, $\mathbf{x_1}$ and $\mathbf{x_2}$. Then $\mathbf{x_1} - \mathbf{x_2}$ would be a $2s$-sparse solution to the homogeneous system $A\mathbf{x} = \mathbf{0}$. A $2s$-sparse solution to the homogeneous equation $A\mathbf{x} = \mathbf{0}$ requires that some $2s$ columns of A be linearly dependent. Unless A has $2s$ linearly dependent columns there can be only one s-sparse solution.

The solution to the reconstruction problem is simple. If the matrix A has at least $2s$

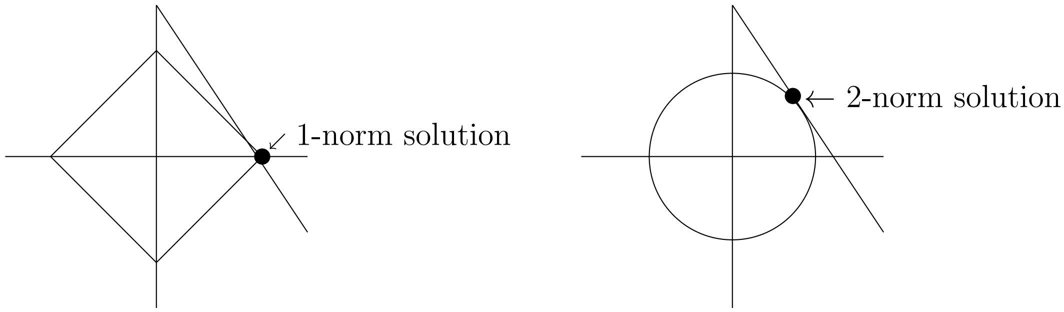

Figure 10.3: Illustration of minimum 1-norm and 2-norm solutions.

rows and the entries of A were selected at random from a standard Gaussian, then with probability one, no set of $2s$ columns will be linearly dependent. We can see this by noting that if we first fix a subset of $2s$ columns and then choose the entries at random, the probability that this specific subset is linearly dependent is the same as the probability that $2s$ random Gaussian vectors in a $2s$-dimensional space are linearly dependent, which is zero.[42] So, taking the union bound over all $\binom{d}{2s}$ subsets, the probability that any one of them is linearly dependent is zero.

The above argument shows that if we choose $n = 2s$ and pick entries of A randomly from a Gaussian, with probability one there will be a unique s-sparse solution. Thus, to solve for $\mathbf{x}$ we could try all $\binom{d}{s}$ possible locations for the nonzero elements in $\mathbf{x}$ and aim to solve $A\mathbf{x} = \mathbf{b}$ over just those s columns of A: any one of these that gives a solution will be the correct answer. However, this takes time $\Omega(d^s)$ which is exponential in s. We turn next to the topic of efficient algorithms, describing a polynomial-time optimization procedure that will find the desired solution when n is sufficiently large and A is constructed appropriately.

10.2.2 Efficiently Finding the Unique Sparse Solution

To find a sparse solution to $A\mathbf{x} = \mathbf{b}$, one would like to minimize the zero norm $\|\mathbf{x}\|_0$ over $\{\mathbf{x}|A\mathbf{x} = \mathbf{b}\}$, i.e., minimize the number of nonzero entries. This is a computationally hard problem. There are techniques to minimize a convex function over a convex set, but $||\mathbf{x}||_0$ is not a convex function, and with no further assumptions, it is NP-hard. With this in mind, we use the one-norm as a proxy for the zero-norm and minimize the one-norm $\|\mathbf{x}\|_1 = \sum_i |x_i|$ over $\{\mathbf{x}|A\mathbf{x} = \mathbf{b}\}$. Although this problem appears to be nonlinear, it can be solved by linear programming by writing $\mathbf{x} = \mathbf{u} - \mathbf{v}$, $\mathbf{u} \geq 0$, and $\mathbf{v} \geq 0$, and minimizing the linear function $\sum\limits_i u_i + \sum\limits_i v_i$ subject to $A\mathbf{u}$-$A\mathbf{v}$=$\mathbf{b}$, $\mathbf{u} \geq 0$, and $\mathbf{v} \geq 0$.

[42]This can be seen by selecting the vectors one at a time. The probability that the i^{th} new vector lies fully in the lower dimensional subspace spanned by the previous $i-1$ vectors is zero, and so by the union bound the overall probability is zero.

We now show if the columns of the n by d matrix A are unit length almost orthogonal vectors with pairwise dot products in the range $(-\frac{1}{2s}, \frac{1}{2s})$ that minimizing $\|\mathbf{x}\|_1$ over $\{\mathbf{x}|A\mathbf{x} = \mathbf{b}\}$ recovers the unique s-sparse solution to $A\mathbf{x}$=$\mathbf{b}$. The ij^{th} element of the matrix A^TA is the cosine of the angle between the i^{th} and j^{th} columns of A. If the columns of A are unit length and almost orthogonal, A^TA will have ones on its diagonal and all off diagonal elements will be small. By Theorem 2.8, if A has $n = s^2 \log d$ rows and each column is a random unit-length n-dimensional vector, with high probability all pairwise dot-products will have magnitude less than $\frac{1}{2s}$ as desired.[43] Here, we use $s^2 \log d$, a larger value of n compared to the existence argument in Section 10.2.1, but now the algorithm is computationally efficient.

Let $\mathbf{x_0}$ denote the unique s-sparse solution to $A\mathbf{x} = \mathbf{b}$ and let $\mathbf{x_1}$ be a solution of smallest possible one-norm. Let $\mathbf{z} = \mathbf{x_1} - \mathbf{x_0}$. We now prove that $\mathbf{z} = \mathbf{0}$ implying that $\mathbf{x_1} = \mathbf{x_0}$. First, $A\mathbf{z} = A\mathbf{x_1} - A\mathbf{x_0} = \mathbf{b} - \mathbf{b} = \mathbf{0}$. This implies that $A^TA\mathbf{z} = \mathbf{0}$. Since each column of A is unit length, the matrix A^TA has ones on its diagonal. Since every pair of distinct columns of A has dot-product in the range $(-\frac{1}{2s}, \frac{1}{2s})$, each off-diagonal entry in A^TA is in the range $(-\frac{1}{2s}, \frac{1}{2s})$. These two facts imply that unless $\mathbf{z} = \mathbf{0}$, every entry in $\mathbf{z}$ must have absolute value less than $\frac{1}{2s}||\mathbf{z}||_1$. If the j^{th} entry in $\mathbf{z}$ had absolute value greater than or equal to $\frac{1}{2s}||\mathbf{z}||_1$, it would not be possible for the j^{th} entry of $A^TA\mathbf{z}$ to equal 0 unless $||\mathbf{z}||_1 = 0$.

Finally let S denote the support of $\mathbf{x_0}$, where $|S| \leq s$. We now argue that $\mathbf{z}$ must have at least half of its ℓ_1 norm inside of S, i.e., $\sum_{j\in S} |z_j| \geq \frac{1}{2}||\mathbf{z}||_1$. This will complete the argument because it implies that the average value of $|z_j|$ for $j \in S$ is at least $\frac{1}{2s}||\mathbf{z}||_1$, which as shown above is only possible if $||\mathbf{z}||_1 = 0$. Let t_{in} denote the sum of the absolute values of the entries of $\mathbf{x_1}$ in the set S, and let t_{out} denote the sum of the absolute values of the entries of $\mathbf{x_1}$ outside of S. So, $t_{in} + t_{out} = ||\mathbf{x_1}||_1$. Let t_0 be the one-norm of $\mathbf{x_0}$. Since $\mathbf{x_1}$ is the minimum one norm solution, $t_0 \geq t_{in} + t_{out}$, or equivalently $t_0 - t_{in} \geq t_{out}$. But $\sum_{j\in S} |z_j| \geq t_0 - t_{in}$ and $\sum_{j\notin S} |z_j| = t_{out}$. This implies that $\sum_{j\in S} |z_j| \geq \sum_{j\notin S} |z_j|$, or equivalently, $\sum_{j\in S} |z_j| \geq \frac{1}{2}||\mathbf{z}||_1$, which as noted above implies that $||\mathbf{z}||_1 = 0$, as desired. ∎

To summarize, we have shown the following theorem and corollary.

Theorem 10.3 *If matrix A has unit-length columns $\mathbf{a_1}, \ldots, \mathbf{a_d}$ and the property that $|\mathbf{a_i} \cdot \mathbf{a_j}| < \frac{1}{2s}$ for all $i \neq j$, then if the equation $A\mathbf{x} = \mathbf{b}$ has a solution with at most s nonzero coordinates, this solution is the unique minimum 1-norm solution to $A\mathbf{x} = \mathbf{b}$.*

Corollary 10.4 *For some absolute constant c, if A has n rows for $n \geq cs^2 \log d$ and each column of A is chosen to be a random unit-length n-dimensional vector, then with high probability A satisfies the conditions of Theorem 10.3 and therefore if the equation $A\mathbf{x} = \mathbf{b}$ has a solution with at most s nonzero coordinates, this solution is the unique minimum 1-norm solution to $A\mathbf{x} = \mathbf{b}$.*

[43] Note that the roles of "n" and "d" are reversed here compared to Theorem 2.8.

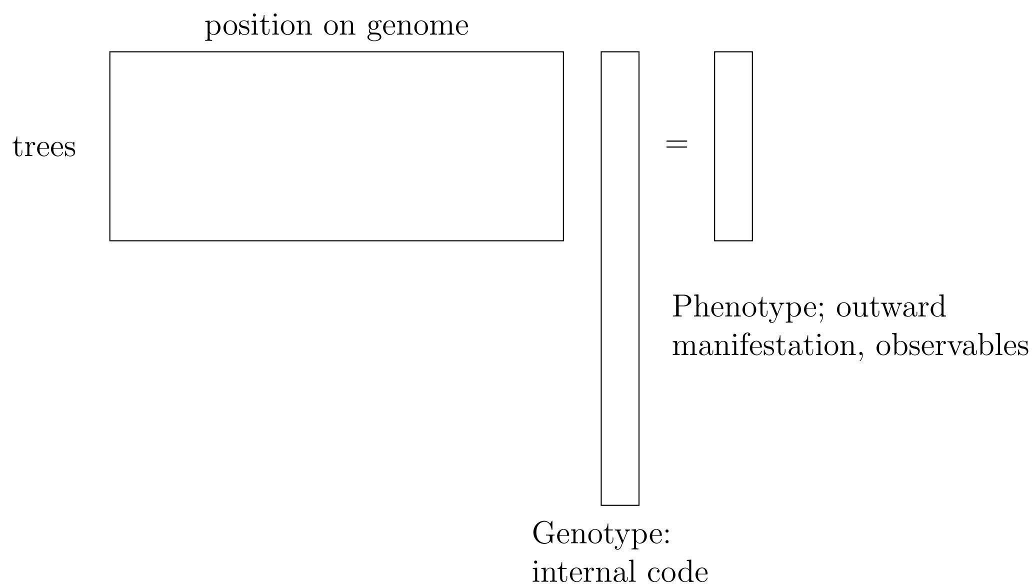

Figure 10.4: The system of linear equations used to find the internal code for some observable phenomenon.

The condition of Theorem 10.3 is often called *incoherence* of the matrix A. Other more involved arguments show that it is possible to recover the sparse solution using one-norm minimization for a number of rows n as small as $O(s \log(ds))$.

10.3 Applications

10.3.1 Biological

There are many areas where linear systems arise in which a sparse solution is unique. One is in plant breeding. Consider a breeder who has a number of apple trees and for each tree observes the strength of some desirable feature. He wishes to determine which genes are responsible for the feature so he can crossbreed to obtain a tree that better expresses the desirable feature. This gives rise to a set of equations $A\mathbf{x} = \mathbf{b}$ where each row of the matrix A corresponds to a tree and each column to a position on the genone. See Figure 10.4. The vector $\mathbf{b}$ corresponds to the strength of the desired feature in each tree. The solution $\mathbf{x}$ tells us the position on the genone corresponding to the genes that account for the feature. It would be surprising if there were two small independent sets of genes that accounted for the desired feature. Thus, the matrix should have a property that allows only one sparse solution.