Supplementary material to “Reconstructing (super)trees from data sets with missing distances: Not all is lost”

George Kettleborough[*], Jo Dicks[†], Ian N. Roberts[‡], and Katharina T. Huber,[§]

November 14, 2014

Corresponding Author Katharina T. Huber, School of Computing Sciences, University of East Anglia, Norwich, United Kingdom; E-mail: katharina.huber@cmp.uea.ac.uk.

1 Introduction

The supplementary material is based on “Reconstructing (super)trees from data sets with missing distances: Not all is lost” by G. Kettleborough, J. Dicks, I. N. Roberts and K. T. Huber. In the next section, we use some of the terminology introduced in its Appendix to give a detailed description of Lasso. In Section 2, we present details concerning our simulation study and some auxiliary results. In Section 3, we present the Neighbour Joining tree for the yeast data set. Finally, in Section 4, we present the Lasso trees for the wheat data sets A and B, together with the Q-matrix for data set A.

1.1 A detailed description of the Lasso algorithm

Given a partial distance matrix D on some taxa set X, Lasso heuristically finds a subset Y of taxa of X as large as possible so that the equidistant tree returned by it is uniquely determined by the available distances on Y with regards to topology and edge-weighting. We next present some detail concerning Lasso.

For ease of presentation, assume from now on that we have already carried out $q \geq 0$ iterations and that C is the selected *connected component* of $\Gamma(\mathcal{L}_{D'})^w$, that is, one of the connected graphs that make up $\Gamma(\mathcal{L}_{D'})^w$ where w is an edge-weighting. Note that for $q = 0$ we may assume that C is $\Gamma(\mathcal{L}_{D'})^w$ itself as connectedness of the graph $\Gamma(\mathcal{L}_{D'})$ and thus of $\Gamma(\mathcal{L}_{D'})^\omega$ is a necessary condition for a Z-tree to be topologically lassoed by a set of cords on some non-empty set Z [Huber and Popescu, 2013]. However it should be noted that $\Gamma(\mathcal{L}_{D'})^w$ may become disconnected during successive repetitions. In that case, we exploit the fact that in each iteration step an equidistant tree is grown heuristically from (hopefully all) the vertices of a connected component of the graph $\Gamma(\mathcal{L}_{D'})^w$ generated in the previous step. Put differently, we choose a connected component of $\Gamma(\mathcal{L}_{D'})^w$ such that, over all connected components of $\Gamma(\mathcal{L}_{D'})^w$, the leaf set of the tree grown from it is as large as possible (where we break ties randomly). Other methods of component choice are conceivable though (see Methods).

Note that Lasso terminates if the selected connected component consists of just one vertex. Also note that to help mitigate against a poor choice of a connected component which might yield an equidistant tree on a small number of taxa of X, Lasso returns the tree that connects the most

[*] School of Computing Sciences, University of East Anglia, Norwich, United Kingdom

[†] National Collection of Yeast Cultures, Institute of Food Research, Norwich Research Park, Norwich, NR4 7UA, United Kingdom

[‡] National Collection of Yeast Cultures, Institute of Food Research, Norwich Research Park, Norwich, NR4 7UA, United Kingdom

[§] School of Computing Sciences, University of East Anglia, Norwich, United Kingdom taxa in X (and its associated strong lasso) found over p independent runs, where p is a parameter that is currently set to ten.

To simplify the description of the remaining details of the q-th iteration step, let m denote the minimal edge weight over all edges of C. Then C is transformed into an unweighted graph C' in which first all edges except those with weight not m are removed and then the weights of the remaining edges are ignored.

Lasso now chooses a connected component S of C and a *suitable* clique K of S (see Methods for an informal description and the Appendix for a formal definition) and grows an equidistant tree using the vertex set of that clique. To make this more precise define for any equidistant tree (T', ω') with leaf set Z the *root height* of T' to be $D_{(T',\omega')}(x, y)/2$ for any two elements x and y of Z for which the path joining them crosses the root of T'. Let G be a graph consisting of $|X|$ isolated vertices each of which is labelled by an element in X. For the purpose of growing a tree it will be useful to view each of them as an equidistant tree with height zero.

Now, let K denote a suitable clique of S with at least two vertices found by Lasso (where we break ties randomly). If no such suitable clique exists then we terminate and save the found equidistant tree(s) and their associated strong lasso(s). Otherwise, to obtain a new distance matrix D' on a smaller set X' which, for example, ensures that Lasso terminates, we first discard all vertices in C that are not contained in K and then replace all vertices of K by a new vertex u. Next, we define D to be the distance matrix on X that assigns to any x and y contained in X the value $D(x, y)$ if x and y in $X' - \{u\}$, zero if $x = y = u_m$ and the value $D^*(x, y)$ if either $x = u$ or $y = u$, where D^* is a distance such as the one described in the Methods section.

To find the equidistant tree (U, ω) that Lasso grows from X in this repetition step, let l denote the size of the vertex set of K and let $(T_1, \omega_1), \ldots, (T_l, \omega_l)$ denote the equidistant trees with leaves in X found in the previous repetition steps such that the vertex set of K comprises of the roots ρ_i of T_i, $1 \leq i \leq l$. Note that some of those trees might just be elements in X. Then to obtain U, we first add a new vertex to G labelled u and then join every root ρ_i, $1 \leq i \leq l$, via an edge with u making U a tree with root u and leaves contained in X.

To obtain the equidistant edge-weighting ω for U, assume for all $i \in \{1, \ldots, l\}$ that the root height h_i of the tree (T_i, ω_i) was computed in one of the previous repetition steps. Note that, by definition, there must exist leaves u and v with distance value $D(u, v) = m$ such that u lies on the path from u to v in U_m. Then we define ω to be the map that assigns to every edge e of U that is also contained in some tree T_i, $1 \leq i \leq l$, its weight under ω_i and the weight $m/2 - h_i$ if e contains u and the root ρ_i where h_i is the root height of T_i, $1 \leq i \leq l$. Since for all $i \in \{1, \ldots, l\}$ the trees (T_i, ω_i) are equidistant, it is straightforward to see that ω_m is an equidistant edge-weighting for U and that the height of the tree (U, ω) is $m/2$.

Finally, we return to finding a connected component for the new graph $\Gamma(\mathcal{L}_{D'})^w$ for D'. Once the aforementioned termination criterion is satisfied, the found equidistant tree and its strong lasso is saved and the next run is started. Lasso stops once all p (currently set to 10) runs have been completed and returns the equidistant tree and its underpinning strong lasso.

2 Simulation studies for missing data

In this section we present the pseudo-code version of one of the algorithms we used to generate for the random trees underpinning our simulation study. In addition, we also use simple statistical measures to shed light onto the sizes of the supporting strong lassos generated by Lasso. We start with the pseudo-code for our algorithm for generating random trees with maximum vertex out-degree k. For this we proceed as before and view an element in X as a tree of height zero.

Tables S1 - S3 provide simple statistical measures for the three tree types that we considered in the simulation part of the paper. The first three columns concern the normalized Robinson Foulds distance and give the mean, min and max values for that distance. The remaining three columns give the number of cords used, where again we present our results in terms of the mean, min and max, denoted by meanc, minc and maxc, respectively.

Algorithm 1 Random tree generation

```
Input: A set $X$ and an integer $k$.
Output: An equidistant tree on $X$ with maximum out-degree $k$.
1. Choose an integer $p$ in $\{2, \dots, \min(|X|, k)\}$ and some subset $C$ of $X$ of size $p$.
2. Construct a tree $T$ with root $\rho_T$ by attaching each element $c$ in $C$ via an edge to $\rho_T$ and
   setting the weight of that edge to $1 + \max_{x \in X} \mathit{height}(x) - \mathit{height}(c)$.
3. Put $X := (X - C) \cup \{\rho_T\}$.
4. If $|X| > 1$ go to step 1, otherwise return $T$ and its equidistant edge-weighting.
```

P_{miss}	mean	min	max	meanc	minc	maxc
0.0	0.000	0.000	0.000	2016.000	2016	2016
1.0	0.000	0.000	0.000	2015.770	2014	2016
5.0	0.011	0.000	0.095	2003.480	1945	2015
10.0	0.029	0.000	0.190	1976.640	1814	2005
20.0	0.099	0.000	0.587	1849.600	475	1944
30.0	0.261	0.000	0.762	1445.220	255	1842

Table S1: Normalized Robinson Foulds distances for the balanced trees and the sizes of the supporting strong lassos.

P_{miss}	mean	min	max	meanc	minc	maxc
0.0	0.000	0.000	0.000	2016.000	2016	2016
1.0	0.000	0.000	0.000	2015.760	2013	2016
5.0	0.005	0.000	0.143	2007.850	1949	2016
10.0	0.024	0.000	0.270	1982.360	1869	2003
20.0	0.110	0.000	0.492	1842.460	672	1954
30.0	0.187	0.000	0.635	1670.370	317	1859

Table S2: Normalized Robinson Foulds distances for the Yule-Harding trees and the sizes of the supporting strong lassos.

P_{miss}	mean	min	max	meanc	minc	maxc
0.0	0.000	0.000	0.000	2016.000	2016	2016
1.0	0.000	0.000	0.000	2015.780	2014	2016
5.0	0.000	0.000	0.000	2011.090	2003	2016
10.0	0.020	0.000	1.000	1994.720	1933	2004
20.0	0.040	0.000	1.000	1931.610	1864	1954
30.0	0.110	0.000	1.000	1829.440	1769	1861

Table S3: Normalized Robinson Foulds distances for the caterpillar trees and the sizes of the supporting strong lassos.

3 Yeast example

In this section we present the Neighbor Joining tree (Fig. S4) estimated from SNP and pSNP sequence variation within the ribosomal DNA regions of 26 strains of the wild yeast *Saccharomyces paradoxus*, together with *Saccharomyces cerevisiae* reference strain S288c, taken from West et al. [2014].

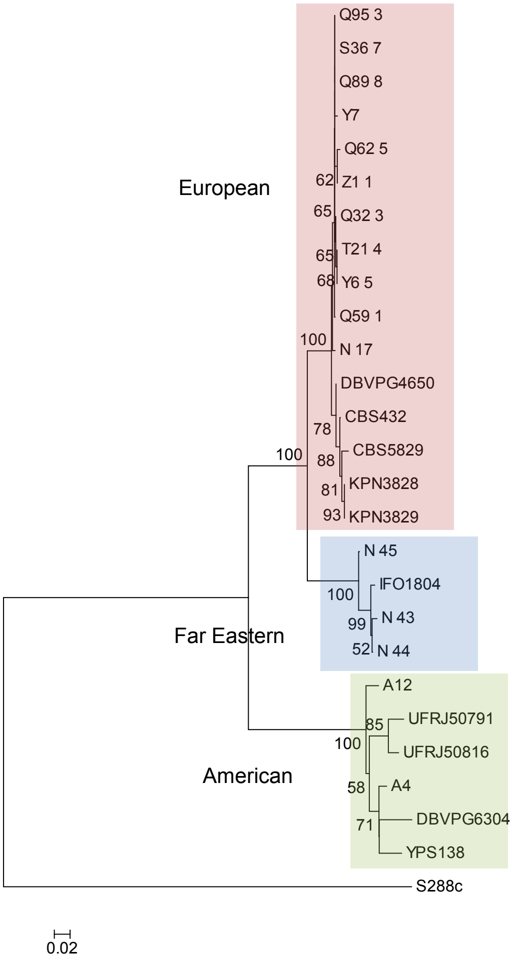

Figure S4: The Neighbor Joining tree for the *Saccharomyces paradoxus* dataset

4 Wheat example

In this section, we present the Lasso trees for the two overlapping datasets that we considered in the supertree example. The first tree (Fig. S5) is on dataset A, which is a subset of a dataset developed within the GEDIFLUX EU Framework V project [Reeves et al., 2004]. The number of founder populations estimated in an ADMIXTURE analysis of this dataset was found to be $K = 3$. Each accession is therefore colored within the Lasso tree according to which of these three populations (red, green and brown) is inferred to have contributed most to its marker scores.

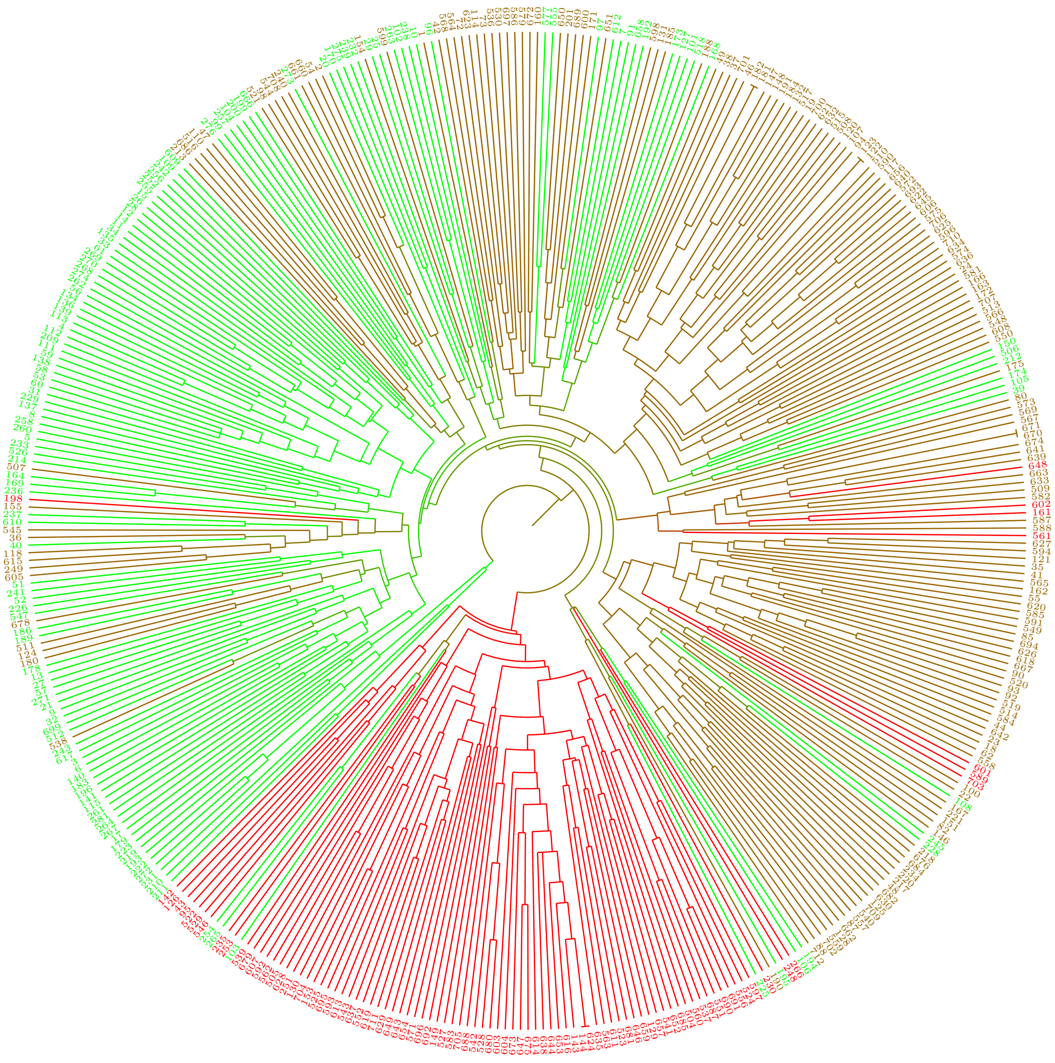

Figure S5: The Lasso tree for wheat dataset A.

The next Lasso tree (Fig. S6) is on dataset B, which is a subset of a wheat dataset that was originally studied in Sayar-Turet et al. [2011]. Again the color coding (red, green, brown and blue) indicates the membership of a wheat accession within one of four founder populations, as determined by a STRUCTURE analysis within the original study. Note that accessions 69, 74 and 121 are coloured black, as 69 was ambiguously grouped in the original study and the remaining two accessions were not included in the STRUCTURE analysis.

Finally, we present in Table S7 the Q-matrix produced within the ADMIXTURE analysis of dataset A.

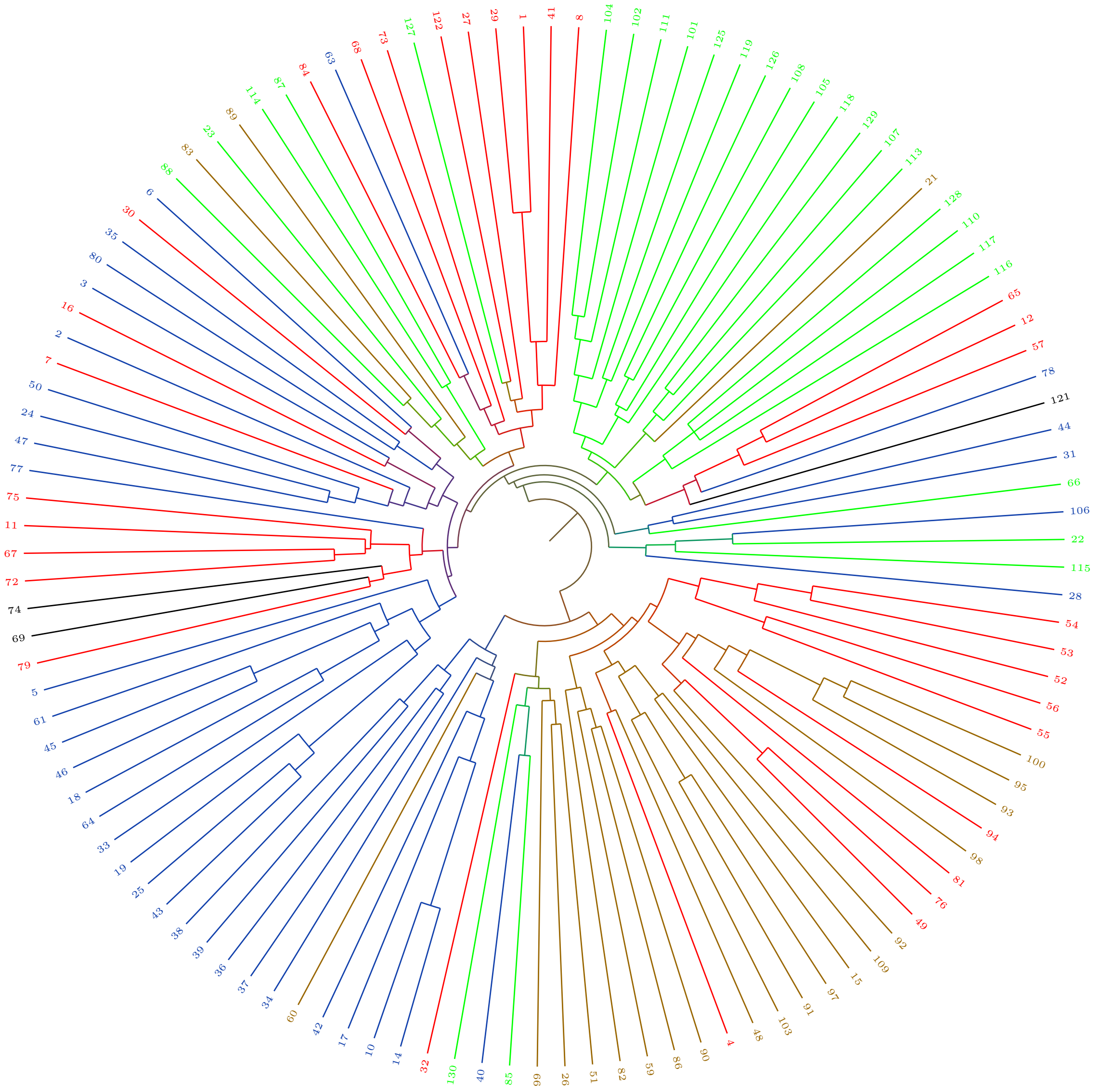

Figure S6: The LASSO tree for wheat dataset B.

References

Katharina T. Huber and Andrei-Alin Popescu. Lassoing and corralling rooted phylogenetic trees. *Bulletin of Mathematical Biology*, 75(3):444–465, 2013. ISSN 0092-8240.

J C Reeves, E Chiapparino, P Donini, M Ganal, J Guiard, S Hamrit, M Heckenberger, X-Q Huang, M Van Kaauwen, E Kochieva, R Koebner, J R Law, V Lea, V Le Clerc, T Van der Lee, F Leigh, G Van der Linden, L Malysheva, A E Melchinger, S Orford, J C Reif, M Röder, A Schulman, B Vosman, C Van der Wiel, M Wolf, and D Zhang. Changes over time in the genetic diversity of four major European crops: a report from the Gediflux Framework 5 project. *Proceedings of the XVIIth EUCARPIA General Congress, Tulln, Austria, 811 September 2004*, pages 3–7, 2004.

M Sayar-Turet, S Dreisigacker, H-J Braun, A Hede, R MacCormack, and LA Boyd. Genetic variation within and between winter wheat genotypes from turkey, kazakhstan and europe as determined by nbs-profiling. *Genome*, 54:419–430, 2011.

Claire West, Stephen A. James, Robert P. Davey, Jo Dicks, and Ian N. Roberts. Ribosomal DNA sequence heterogeneity reflects intraspecies phylogenies and predicts genome structure in two contrasting yeast species. *Systematic Biology*, 63(4):543–554, 2014.

	Pop. group			
A	1	2	3	P
1	0.00001	0.30351	0.69648	3
2	0.00001	0.36723	0.63276	3
3	0.03078	0.53630	0.43292	2
4	0.10535	0.89464	0.00001	2
5	0.00001	0.99998	0.00001	2
6	0.00001	0.99998	0.00001	2
7	0.00001	0.99998	0.00001	2
8	0.00001	0.69542	0.30458	2
9	0.00001	0.84732	0.15267	2
10	0.00001	0.55922	0.44077	2
11	0.00001	0.99998	0.00001	2
12	0.00001	0.99640	0.00359	2
13	0.00001	0.62969	0.37030	2
14	0.00001	0.99998	0.00001	2
15	0.00001	0.99998	0.00001	2
17	0.00001	0.99998	0.00001	2
18	0.00001	0.00001	0.99998	3
19	0.00001	0.30844	0.69156	3
20	0.00001	0.66604	0.33395	2
21	0.00001	0.99849	0.00150	2
22	0.00001	0.02259	0.97740	3
23	0.00001	0.00001	0.99998	3
24	0.00001	0.37111	0.62888	3
25	0.07127	0.70919	0.21954	2
26	0.00001	0.99998	0.00001	2
27	0.00001	0.84756	0.15243	2
28	0.00001	0.80176	0.19823	2
29	0.00001	0.62673	0.37326	2
31	0.00001	0.99998	0.00001	2
32	0.00001	0.95254	0.04745	2
33	0.00001	0.99998	0.00001	2
34	0.00001	0.99998	0.00001	2
35	0.00001	0.00001	0.99998	3
36	0.04707	0.43700	0.51594	3
37	0.00001	0.44356	0.55643	3
38	0.00001	0.41039	0.58960	3
39	0.00001	0.61779	0.38220	2
40	0.21059	0.58776	0.20165	2
41	0.00001	0.00001	0.99998	3
42	0.11061	0.39008	0.49932	3
43	0.99998	0.00001	0.00001	1
44	0.00001	0.21931	0.78068	3
46	0.00001	0.21305	0.78694	3
47	0.00001	0.32656	0.67343	3
48	0.00001	0.43070	0.56929	3
49	0.00001	0.01548	0.98451	3
51	0.00001	0.99998	0.00001	2
52	0.00001	0.75560	0.24439	2
53	0.00001	0.86397	0.13602	2
54	0.01516	0.34984	0.63500	3
55	0.00001	0.33053	0.66946	3
56	0.00001	0.96539	0.03460	2
57	0.00001	0.89313	0.10686	2
58	0.00001	0.99998	0.00001	2
59	0.00001	0.99998	0.00001	2
60	0.00001	0.88908	0.11091	2
61	0.00001	0.99998	0.00001	2
62	0.00001	0.17839	0.82160	3
72	0.00001	0.38229	0.61770	3
73	0.04173	0.01423	0.94404	3
80	0.00001	0.41365	0.58634	3
82	0.00001	0.26591	0.73409	3
85	0.00001	0.00001	0.99998	3
87	0.22056	0.23298	0.54647	3
88	0.00001	0.00001	0.99998	3
90	0.00001	0.08926	0.91073	3
91	0.00001	0.56271	0.43728	2
92	0.00001	0.00001	0.99998	3
93	0.00001	0.00001	0.99998	3
94	0.06212	0.13356	0.80433	3
95	0.86964	0.13035	0.00001	1
96	0.00001	0.58947	0.41052	2
97	0.00001	0.52821	0.47178	2
98	0.00001	0.31755	0.68244	3
99	0.28084	0.69182	0.02734	2
100	0.00001	0.28464	0.71535	3
101	0.00001	0.58216	0.41783	2
102	0.39707	0.60292	0.00001	2
103	0.34502	0.65497	0.00001	2
104	0.33596	0.66403	0.00001	2
105	0.07834	0.92165	0.00001	2
106	0.17998	0.66613	0.15389	2
107	0.00001	0.00001	0.99998	3
108	0.00001	0.61082	0.38918	2
109	0.00001	0.77349	0.22650	2
110	0.08135	0.35809	0.56056	3
111	0.00001	0.99998	0.00001	2
112	0.09763	0.90236	0.00001	2
113	0.12807	0.87192	0.00001	2
114	0.16650	0.16472	0.66879	3
115	0.00001	0.83956	0.16043	2
117	0.00227	0.00001	0.99772	3
118	0.00001	0.17023	0.82976	3
120	0.00001	0.00001	0.99998	3
121	0.00001	0.14558	0.85441	3
122	0.00001	0.00001	0.99998	3
123	0.00001	0.00001	0.99998	3
124	0.00001	0.42802	0.57197	3
125	0.32844	0.00001	0.67155	3
126	0.98799	0.01201	0.00001	1
127	0.02806	0.48659	0.48535	2
128	0.24629	0.19782	0.55589	3
129	0.81929	0.18070	0.00001	1
130	0.99998	0.00001	0.00001	1
131	0.29728	0.28028	0.42243	3
132	0.00001	0.33357	0.66642	3
134	0.00001	0.99998	0.00001	2
135	0.00001	0.99998	0.00001	2
136	0.00001	0.99998	0.00001	2
137	0.00001	0.99998	0.00001	2
138	0.00001	0.99998	0.00001	2
139	0.12567	0.87433	0.00001	2
140	0.00001	0.99998	0.00001	2
141	0.00001	0.69608	0.30391	2
142	0.61378	0.38621	0.00001	1
143	0.99998	0.00001	0.00001	1
144	0.99998	0.00001	0.00001	1
146	0.00001	0.38039	0.61960	3
147	0.00001	0.18673	0.81326	3
148	0.00001	0.20163	0.79836	3
149	0.67416	0.23291	0.09292	1
150	0.21556	0.46075	0.32369	2
151	0.00001	0.37564	0.62435	3
153	0.99998	0.00001	0.00001	1
154	0.00001	0.28143	0.71856	3
155	0.26789	0.24321	0.48891	3
160	0.17397	0.33051	0.49552	3
161	0.58419	0.14201	0.27379	1
162	0.00001	0.34426	0.65573	3
163	0.05339	0.25135	0.69526	3
164	0.00001	0.71714	0.28285	2
166	0.27307	0.00001	0.72692	3
167	0.00001	0.31596	0.68403	3
168	0.00001	0.51539	0.48460	2
169	0.00001	0.91073	0.08926	2
170	0.00001	0.52673	0.47326	2
171	0.00001	0.32368	0.67631	3
172	0.00001	0.09661	0.90338	3
173	0.00001	0.90174	0.09825	2
174	0.00001	0.58955	0.41044	2
175	0.00001	0.35805	0.64194	3
178	0.00001	0.99998	0.00001	2
179	0.00001	0.87101	0.12898	2
180	0.00001	0.34017	0.65982	3
181	0.00001	0.00001	0.99998	3
182	0.00001	0.00001	0.99998	3
183	0.00001	0.77228	0.22771	2
184	0.00001	0.35639	0.64360	3
185	0.00001	0.49068	0.50931	3
186	0.00001	0.60805	0.39194	2
188	0.00001	0.48983	0.51016	3
189	0.00001	0.52325	0.47674	2
190	0.00001	0.40636	0.59363	3
191	0.00001	0.32358	0.67641	3
192	0.00001	0.79572	0.20428	2
193	0.00001	0.99998	0.00001	2
194	0.36198	0.62757	0.01045	2
195	0.00001	0.58985	0.41014	2
196	0.00001	0.69626	0.30373	2
198	0.34983	0.33705	0.31313	1
199	0.00001	0.95813	0.04187	2
200	0.28075	0.37531	0.34394	2
201	0.00001	0.39177	0.60822	3
209	0.15174	0.84825	0.00001	2
210	0.26572	0.73427	0.00001	2
211	0.16019	0.83980	0.00001	2
212	0.00001	0.99496	0.00503	2
213	0.23136	0.00001	0.76863	3
214	0.26731	0.73268	0.00001	2
217	0.00001	0.28824	0.71175	3
218	0.00001	0.12159	0.87840	3
219	0.00001	0.99998	0.00001	2
220	0.00001	0.35237	0.64762	3
221	0.03872	0.38680	0.57448	3
222	0.00001	0.99998	0.00001	2
223	0.00001	0.81779	0.18220	2
225	0.26078	0.51178	0.22744	2
226	0.00755	0.78754	0.20492	2
228	0.00001	0.51538	0.48461	2
229	0.00001	0.99998	0.00001	2
230	0.43742	0.17500	0.38758	1
231	0.00001	0.84515	0.15484	2
232	0.00001	0.70321	0.29678	2
233	0.00001	0.99998	0.00001	2
234	0.00001	0.35288	0.64711	3
235	0.78895	0.21104	0.00001	1
236	0.00001	0.54401	0.45598	2
237	0.28428	0.37493	0.34079	2
238	0.27471	0.56693	0.15836	2
239	0.00001	0.89236	0.10763	2
240	0.13739	0.18467	0.67794	3
241	0.19304	0.80695	0.00001	2
242	0.10296	0.89703	0.00001	2
243	0.00618	0.70180	0.29203	2
245	0.06447	0.50785	0.42769	2
248	0.41881	0.29130	0.28989	1
249	0.01074	0.20737	0.78189	3
251	0.78891	0.21108	0.00001	1
252	0.00001	0.99998	0.00001	2
253	0.74077	0.25922	0.00001	1
254	0.47335	0.52664	0.00001	2
255	0.00001	0.89852	0.10147	2
256	0.00001	0.73922	0.26077	2
257	0.16807	0.71629	0.11564	2
258	0.00001	0.99998	0.00001	2
260	0.00001	0.84997	0.15002	2
261	0.00001	0.86019	0.13980	2
262	0.00001	0.99998	0.00001	2
263	0.33285	0.64156	0.02559	2
264	0.33780	0.66219	0.00001	2
265	0.46274	0.53725	0.00001	2
266	0.60498	0.13734	0.25768	1
267	0.00001	0.99998	0.00001	2
268	0.00001	0.99998	0.00001	2
269	0.00001	0.99998	0.00001	2
270	0.00001	0.99998	0.00001	2
271	0.00001	0.99998	0.00001	2
272	0.00001	0.99998	0.00001	2
273	0.18756	0.71796	0.09449	2
274	0.00001	0.99998	0.00001	2
275	0.27298	0.68776	0.03926	2
276	0.00001	0.61884	0.38115	2
281	0.21490	0.46562	0.31948	2
504	0.99998	0.00001	0.00001	1
505	0.99998	0.00001	0.00001	1
506	0.00001	0.69461	0.30538	2
507	0.00001	0.44737	0.55262	3
508	0.00001	0.00001	0.99998	3
509	0.38145	0.00001	0.61854	3
510	0.00001	0.00001	0.99998	3
511	0.01156	0.38798	0.60046	3
512	0.06654	0.63771	0.29575	2
513	0.00001	0.00001	0.99998	3
514	0.00001	0.00001	0.99998	3
515	0.00001	0.00001	0.99998	3
516	0.00001	0.00001	0.99998	3
517	0.00001	0.18916	0.81083	3
518	0.00001	0.51456	0.48543	2
519	0.19917	0.00001	0.80082	3
520	0.00001	0.00001	0.99998	3
521	0.00001	0.15654	0.84345	3
522	0.99998	0.00001	0.00001	1
523	0.99998	0.00001	0.00001	1
524	0.90384	0.09615	0.00001	1
525	0.99998	0.00001	0.00001	1
526	0.25991	0.48914	0.25095	2
527	0.66901	0.12104	0.20995	1
528	0.99998	0.00001	0.00001	1
529	0.95457	0.04542	0.00001	1
530	0.07237	0.20721	0.72042	3
536	0.47352	0.00001	0.52647	3
537	0.99998	0.00001	0.00001	1
538	0.00001	0.48918	0.51081	3
539	0.71415	0.28584	0.00001	1
540	0.30638	0.27917	0.41445	3
541	0.99998	0.00001	0.00001	1
542	0.99998	0.00001	0.00001	1
543	0.99998	0.00001	0.00001	1
544	0.00001	0.25447	0.74552	3
545	0.13648	0.37232	0.49120	3
546	0.99998	0.00001	0.00001	1
547	0.01562	0.68386	0.30052	2
548	0.00001	0.00001	0.99998	3
549	0.00001	0.00001	0.99998	3
550	0.16611	0.18693	0.64696	3
551	0.99998	0.00001	0.00001	1
552	0.99998	0.00001	0.00001	1
555	0.34328	0.38922	0.26751	2
556	0.85842	0.14157	0.00001	1
557	0.34320	0.17678	0.48002	3
558	0.00001	0.00001	0.99998	3
559	0.27368	0.24306	0.48326	3
560	0.96531	0.03468	0.00001	1
561	0.58662	0.00523	0.40815	1
562	0.99998	0.00001	0.00001	1
563	0.88187	0.11812	0.00001	1
564	0.00001	0.17881	0.82118	3
565	0.00001	0.00001	0.99998	3
566	0.00001	0.00001	0.99998	3
567	0.26290	0.00001	0.73709	3
568	0.15655	0.18150	0.66196	3
569	0.35664	0.15278	0.49059	3
571	0.82158	0.00001	0.17841	1
572	0.13178	0.00001	0.86821	3
573	0.11799	0.00001	0.88200	3
574	0.00001	0.00001	0.99998	3
575	0.23057	0.00001	0.76942	3
577	0.17678	0.43480	0.38842	2
579	0.00001	0.35363	0.64636	3
581	0.00001	0.24577	0.75422	3
582	0.36027	0.00001	0.63972	3
583	0.72689	0.27310	0.00001	1
584	0.11601	0.23342	0.65057	3
585	0.00001	0.11241	0.88758	3
586	0.17021	0.18626	0.64354	3
587	0.38411	0.00001	0.61588	3
588	0.28539	0.15519	0.55942	3
589	0.48113	0.20265	0.31621	1
590	0.24861	0.00001	0.75138	3
591	0.00001	0.15029	0.84970	3
592	0.84174	0.01053	0.14773	1
593	0.99998	0.00001	0.00001	1
594	0.05632	0.00001	0.94368	3
596	0.29348	0.00001	0.70651	3
597	0.62543	0.15388	0.22069	1
598	0.00001	0.18742	0.81257	3
599	0.00913	0.43663	0.55424	3
600	0.18295	0.30729	0.50976	3
601	0.61044	0.01399	0.37557	1
602	0.59823	0.00001	0.40176	1
603	0.54482	0.00001	0.45517	1
604	0.99998	0.00001	0.00001	1
605	0.00001	0.30103	0.69896	3
606	0.00001	0.00001	0.99998	3
607	0.69393	0.30606	0.00001	1
608	0.09535	0.17768	0.72697	3
609	0.36009	0.39175	0.24816	2
610	0.34696	0.36552	0.28752	2
611	0.99998	0.00001	0.00001	1
612	0.06095	0.48844	0.45062	2
613	0.99998	0.00001	0.00001	1
614	0.86501	0.13499	0.00001	1
615	0.00001	0.00001	0.99998	3
616	0.99998	0.00001	0.00001	1
617	0.17138	0.48245	0.34618	2
618	0.00001	0.00001	0.99998	3
619	0.99998	0.00001	0.00001	1
620	0.00001	0.10718	0.89282	3
621	0.01092	0.00001	0.98907	3
622	0.07301	0.00001	0.92698	3
623	0.18148	0.21840	0.60012	3
624	0.99998	0.00001	0.00001	1
625	0.10730	0.01419	0.87851	3
626	0.14394	0.00515	0.85091	3
627	0.00001	0.00001	0.99998	3
628	0.99998	0.00001	0.00001	1
629	0.99998	0.00001	0.00001	1
630	0.43362	0.00001	0.56638	3
632	0.07831	0.00001	0.92168	3
633	0.44514	0.00001	0.55485	3
634	0.00001	0.22162	0.77837	3
635	0.99998	0.00001	0.00001	1
636	0.00001	0.41116	0.58883	3
637	0.99998	0.00001	0.00001	1
638	0.99998	0.00001	0.00001	1
639	0.33031	0.00001	0.66968	3
640	0.02942	0.00001	0.97057	3
641	0.39143	0.00001	0.60856	3
642	0.30109	0.00001	0.69890	3
643	0.99998	0.00001	0.00001	1
644	0.99998	0.00001	0.00001	1
645	0.11004	0.00001	0.88995	3
646	0.99998	0.00001	0.00001	1
647	0.99998	0.00001	0.00001	1
648	0.50310	0.00001	0.49689	1
649	0.99998	0.00001	0.00001	1
650	0.00001	0.44815	0.55184	3
651	0.25560	0.29104	0.45336	3
652	0.99998	0.00001	0.00001	1
653	0.89541	0.10458	0.00001	1
654	0.99998	0.00001	0.00001	1
656	0.00001	0.00001	0.99998	3
657	0.99998	0.00001	0.00001	1
658	0.00001	0.63383	0.36617	2
659	0.99998	0.00001	0.00001	1
660	0.10160	0.25297	0.64544	3
661	0.00001	0.18974	0.81025	3
662	0.28094	0.14337	0.57569	3
663	0.31282	0.18426	0.50293	3
666	0.00001	0.59176	0.40823	2
667	0.27644	0.00001	0.72355	3
670	0.21861	0.00001	0.78138	3
671	0.21861	0.00001	0.78138	3
672	0.12556	0.29689	0.57755	3
673	0.85857	0.14142	0.00001	1
674	0.36912	0.00001	0.63087	3
675	0.99998	0.00001	0.00001	1
676	0.07995	0.41153	0.50853	3
678	0.38136	0.12384	0.49481	3
679	0.51868	0.32569	0.15564	1
680	0.99998	0.00001	0.00001	1
681	0.99998	0.00001	0.00001	1
682	0.00001	0.31694	0.68305	3
684	0.31221	0.07764	0.61015	3
685	0.99998	0.00001	0.00001	1
686	0.00001	0.31578	0.68421	3
687	0.99998	0.00001	0.00001	1
688	0.99998	0.00001	0.00001	1
689	0.00001	0.16733	0.83266	3
690	0.99998	0.00001	0.00001	1
691	0.99998	0.00001	0.00001	1
692	0.99998	0.00001	0.00001	1
693	0.04941	0.00001	0.95058	3
694	0.00001	0.19740	0.80259	3
695	0.99998	0.00001	0.00001	1
696	0.99998	0.00001	0.00001	1
697	0.00001	0.27224	0.72775	3
699	0.21015	0.64928	0.14057	2
700	0.06064	0.00001	0.93935	3
701	0.00001	0.28484	0.71515	3
702	0.34261	0.00001	0.65738	3
703	0.51153	0.00001	0.48846	1
704	0.99998	0.00001	0.00001	1
705	0.84011	0.15988	0.00001	1
706	0.09836	0.00001	0.90163	3
707	0.27944	0.34538	0.37518	3
708	0.02142	0.29817	0.68041	3
709	0.35992	0.13996	0.50013	3
710	0.00001	0.00001	0.99998	3
711	0.99998	0.00001	0.00001	1
712	0.23228	0.00001	0.76771	3

Table 1: Table S7: Matrix showing the genetic contribution of the three inferred founder populations to each of the 411 accessions within wheat dataset A, as estimated by the ADMIXTURE software. GEDIFLUX accession numbers are shown in columns denoted by A and the main population group (i.e. that which gave the greatest contribution to each accession) is shown in columns denoted by P.