```
In[◦]:= Clear[m];
Clear[l];
Clear[K];
Clear[I1];
Clear[I2];
Clear[C0];
Clear[C1];
Clear[C2];
Clear[DotC];
Clear[n0];
Clear[fbare];
Clear[ft];
Clear[ξ];
Clear[fGS];
Clear[TabSingle];
Clear[TabAll];
Clear[x];
Clear[a];
Clear[c];
Clear[Delta];
```

Functions

```
In[ ]:= p = 1;
     h = 1;
     Nb = 20;
     I1[p1_, p2_] := Beta[p1 + 1, p2 - p1 - 1] * 2 ^ (p1 - p2 + 1);
     I2[p1_, p2_] := I1[p1, p2 + 1];
     (**)
     C0[n_] := 8 * n^2;
     C1[n_] := 2 (1 + 2 * n * (n + 1) - 3 (2 n + 1));
     C2[n_] := 2 (1 - 2 (n + 1));
     (**)
     DotC[k_, n_, l_, m_] :=
       -Which[n == 0 && m == 0, C1[n] * I1[k + n, k + n + 2 p + 2] + C2[n] * I1[k + n + 1, k + n + 2 p + 2] -
          l (l + 1) I1[k + n, k + n + 2 p], n > 0 && m == 0, C0[n] * I1[k + n - 1, k + n + 2 p + 2] +
          C1[n] * I1[k + n, k + n + 2 p + 2] + C2[n] * I1[k + n + 1, k + n + 2 p + 2] -
          l (l + 1) I1[k + n, k + n + 2 p], n > 0 && m > 0, C0[n] * I1[k + n - 1, k + n + 2 p + 2] +
          C1[n] * I1[k + n, k + n + 2 p + 2] + C2[n] * I1[k + n + 1, k + n + 2 p + 2] -
          l (l + 1) I1[k + n, k + n + 2 p] - m^2 * I2[k + n - 1, k + n + 2 p]];
     (**)
     ComputeY1[l_, m_, n0_] := Module[{MatC, R1, Y1}, Print["Computing Gram Matrix..."];
       MatC = ParallelTable[DotC[k, n, l, m], {k, n0, Nb}, {n, n0, Nb}];
       Print["Computing Cholesky decomposition..."];
       R1 = CholeskyDecomposition[MatC];
       Print["Computing orthogonal coefficients..."];
     Y1 = Inverse[R1]; Return[Y1]]
     (**)
     ExportY1[Nmax_, l_, m_] := Module[{Y1, Y1String, n0}, n0 = If[m == 0, 0, 1];
       Y1 = ComputeY1[l, m, n0];
       Y1String = Table[StringReplace[ToString[Y1[[i, j]], InputForm], {"Sqrt[" :> "sqrt(",
          "]" :> ")"}], {i, 1, Length[Y1[[1, All]]]}, {j, 1, Length[Y1[[All, 1]]]}];
       Export[NotebookDirectory[] <> "GS_coeffs/h_1/l_" <> ToString[l] <>
         "/m_" <> ToString[m] <> "/Y_l_" <> ToString[l] <> "_m_" <>
         ToString[m] <> ".hdf5", Y1String, {"Datasets", "tabY"}]];
     (**)
     AutomaticY1[Nmax_, m_, lmin_, lmax_] := Do[Print["l=", l];
       ExportY1[Nmax, l, m], {l, lmin, lmax}];
     (**)
     AutomaticY1Ell[Nmax_, l_, mmin_, mmax_] := Do[Print["m=", m];
       ExportY1[Nmax, l, m], {m, mmin, mmax}];
```

```
m = 0;
lmin = 0;
lmax = 20;
(**)
AutomaticY1[Nb, m, lmin, lmax]
```