Hapl-o-Mat – Data Preparation

Please refer to gettingStarted, detailedGettingStartedLinux, or detailedGettingStartedWindows for information on how to use Hapl-o-Mat.

Data Preparation

Hapl-o-Mat relies on information on the HLA nomenclature. This information is provided by data files, which we are going to create. As the HLA nomenclature evolves over time, e.g. by finding new alleles or adding new multiple allele codes, it is important to consider to update this information from time to time to allow new alleles to be handled by Hapl-o-Mat. Keep also in mind, that sometimes, rarely, alleles are also removed from the nomenclature. Thus rerunning older analyses can behave differently.

Hapl-o-Mat relies on the following files, which must be placed in the folder "Hapl-o-Mat/data" for Hapl-o-Mat to work:

File name	Description
AllAllelesExpanded.txt	A list of relevant existing HLA alleles with their enclosed more-digit typing resolutions
AlleleList.txt	If your input data in GLS format includes a missing single-locus genotype, it can be replaced by combining all alleles of the same locus from this file. This file is only required, if you are going to use this feature.
Ambiguity.txt	Data for the ambiguity filter
LargeG.txt	A list of G-groups with their enclosed alleles in 8-digit resolution
MultipleAlleleCodes.txt	A list of multiple allele codes and their translation to alleles in 4-digit resolution
P.txt	A list of P-groups with their enclosed alleles in 8-digit resolution
Smallg.txt	A list of g-groups with their enclosed alleles in 8-digit resolution

In the following we are going to create these data files. Enter the folder "prepareData". Everything is going to happen from here.

As the data-processing is a little bit tedious, we provide you with an automated script, see "Automated Way". If you prefer to do it all on your own, head to "Manual Way". The "Automated Way" relies on being able to download files from the internet. That can sometimes be hampered by firewall or proxy settings. If you can download files by different means and just want to skip fiddling with connectivity settings in python, we also provide a "Semi-Automated Way" that does everything for you except downloading files.

Automated Way

Just run the python script "BuildData.py", which does the whole job for you including creating the folder "Hapl-o-Mat/data" and moving the required files there.

Semi-Automated Way

If you have tried the “Automated Way” and the script was not able to run properly to the end but instead threw an error such as “unable to download file” or “connection timeout” then you might be able to still get your data prepared almost automated. Except for the download.

Presuming you have access to the internet and can download data, then to cope with the connection errors you can download the following four files manually: hla_nom_p.txt, hla_nom_g.txt, alpha.v3.zip, and hla_ambigs.xml.zip. You can get these files at the following locations:

1. hla_nom_p.txt
   Go to http://hla.alleles.org/wmda/hla_nom_p.txt to get this file.
2. hla_nom_g.txt
   Go to http://hla.alleles.org/wmda/hla_nom_g.txt to get this file.
3. alpha.v3.zip
   Go to https://bioinformatics.bethematchclinical.org/HLA/alpha.v3.zip to get this file.
4. hla_ambigs.xml.zip
   Go to https://github.com/jrob119/IMGTHLA/raw/Latest/xml/hla_ambigs.xml.zip to get this file.

If you need a more detailed description of what to do, please refer to “Manual Way”, section “Download Data”, steps 1, 2, 3, and 4a).

Once you have downloaded these files, place them in a separate folder of your liking for further reference. Then place a copy of these four files in the directory “Hapl-o-Mat/prepareData”. Please note, that the four files in “Hapl-o-Mat/prepareData” will be removed after data preparation so keeping a copy of them in separate folder of you liking is advised.

After copying the files to “Hapl-o-Mat/prepareData”, just run the script “BuildData.py” (e.g. via “python3 BuildData.py”. The script will realize that these files are already present, skip the download and proceed from there.

Troubleshooting

In rare cases, BuildData.py will not be able to succeed. This happens, if one or more of the files (hla_nom_p.txt, hla_nom_g.txt, alpha.v3.zip, hla_ambigs.xml.zip, g.txt, alpha.v3, ore hla_ambigs.xml) is present in the directory “Hapl-o-Mat/prepareData” but is for any reason incomplete, broken, or corrupted in any way.
This situation can be redeemed by deleting these files, replacing them with newly downloaded copies and running BuildData.py again.

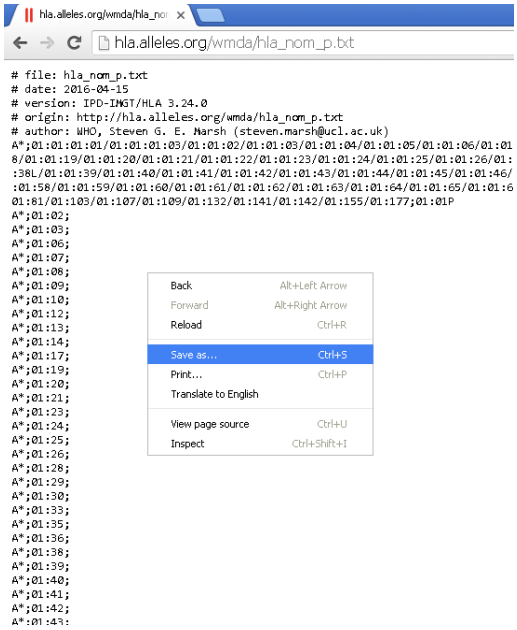
Manual Way

Here, we perform the data preparation step by step.

Download Data

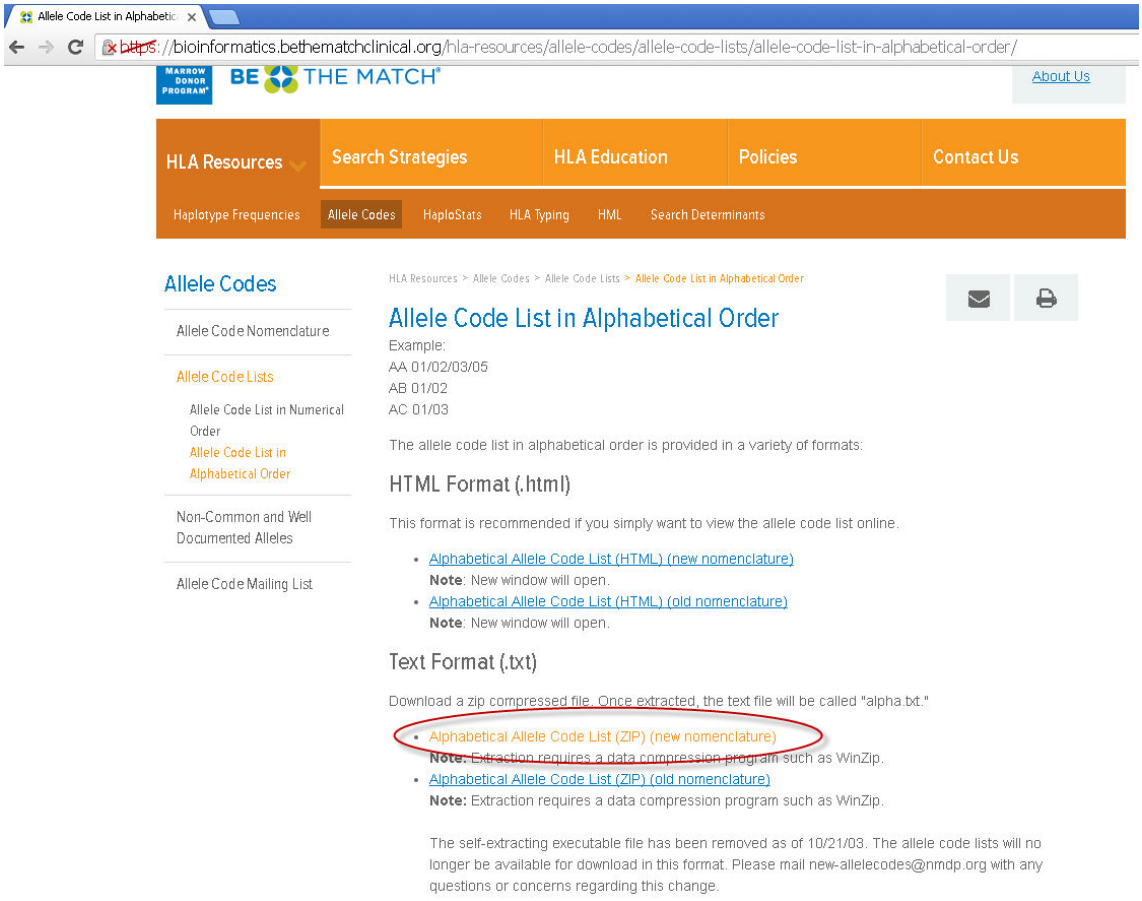
First, we need some input data from the internet. Download the following files:

1) Go to the website http://hla.alleles.org/wmda/hla_nom_p.txt and save the file hla_nom_p.txt by right-clicking and choosing “Save as…”. Move the file “hla_nom_p.txt” to the folder “Hapl-o-Mat/prepareData”.



2) Go to the website http://hla.alleles.org/wmda/hla_nom_g.txt and save the file hla_nom_g.txt (same as in 1)). Move the file “hla_nom_g.txt” to the folder “Hapl-o-Mat/prepareData”.

3) Go to the website https://bioinformatics.bethematchclinical.org/HLA-Resources/Allele-Codes/Allele-Code-Lists/Allele-Code-List-in-Alphabetical-Order/. Click on "Alphabetical Allele Code List (ZIP) (new nomenclature)" and save alpha.v3.zip.

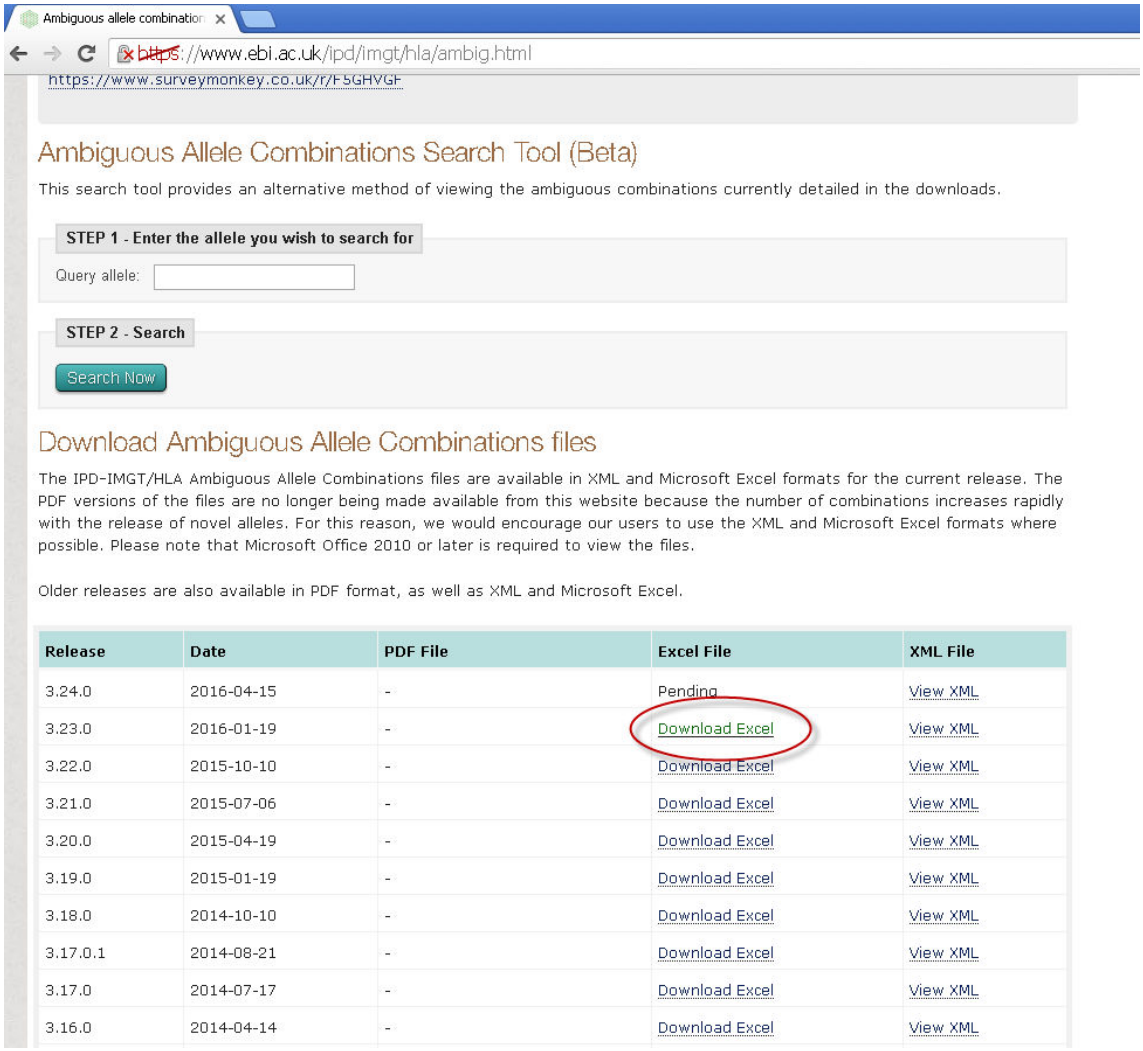


Extract the archive alpha.v3.zip. This should be straightforward in Windows. Using a Terminal under Linux you can use the command “unzip alpha.v3.zip”. You can remove the archive “alpha.v3.zip” afterwards. We only need the file “alpha.v3.txt”. Move it to the folder “Hapl-o-Mat/prepareData”.

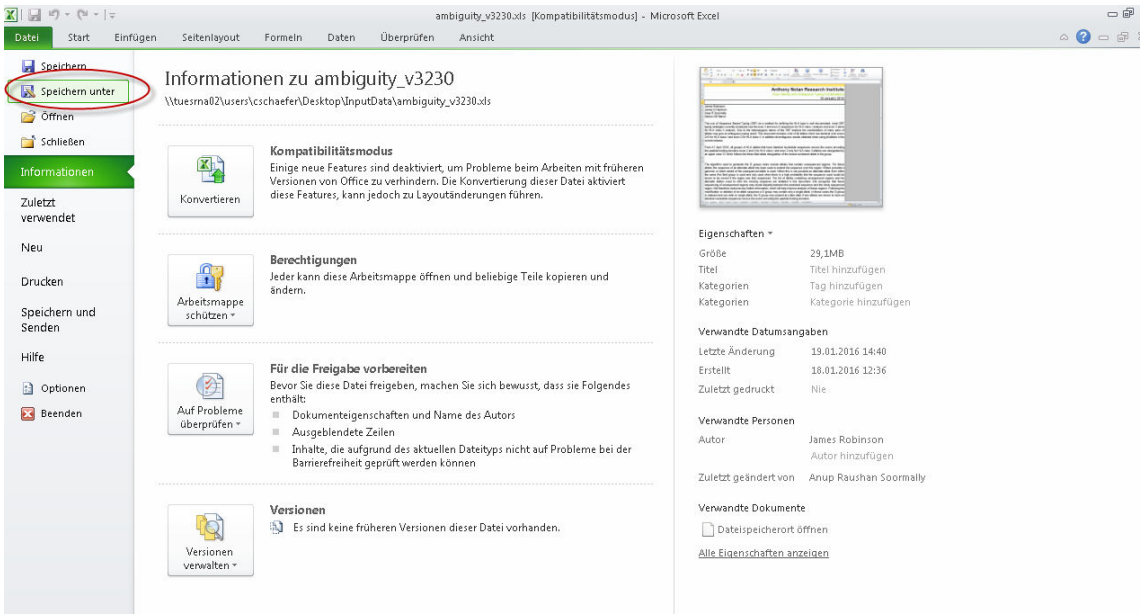
4) You have two options to download the next file. The first approach is simpler and quicker.

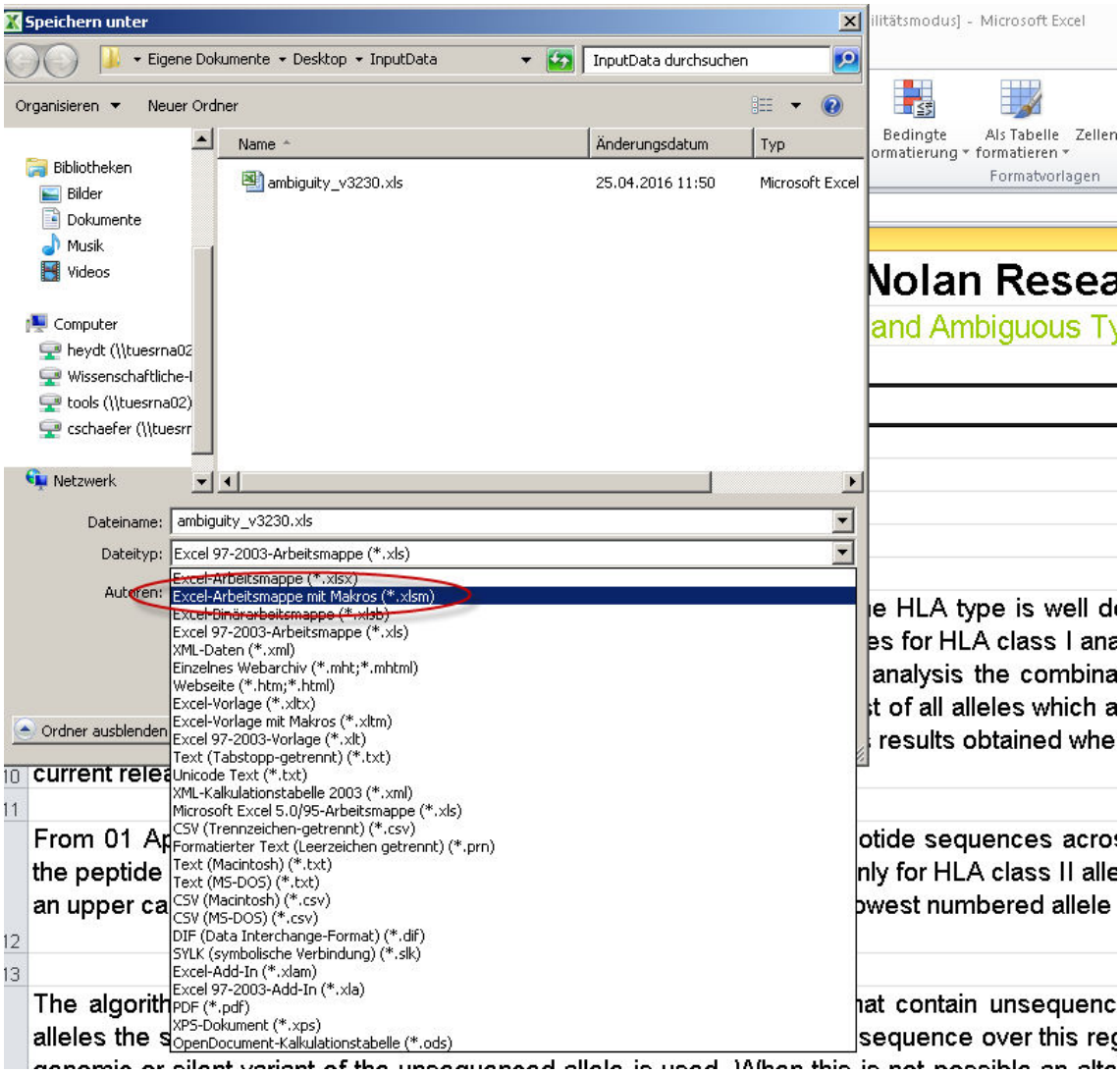
   a) Download the file https://github.com/jrob119/IMGTHLA/raw/Latest/xml/hla_ambigs.xml.zip. Extract it as in 3) including removing the archive. Move the file hla_ambigs.xml to folder “Hapl-o-Mat/prepareData”.

   b) Go to the website https://www.ebi.ac.uk/ipd/imgt/hla/ambig.html. Click on "Download Excel" for the wanted release (usually the latest) and save ambiguity_v<>.xls (replace <> by version).



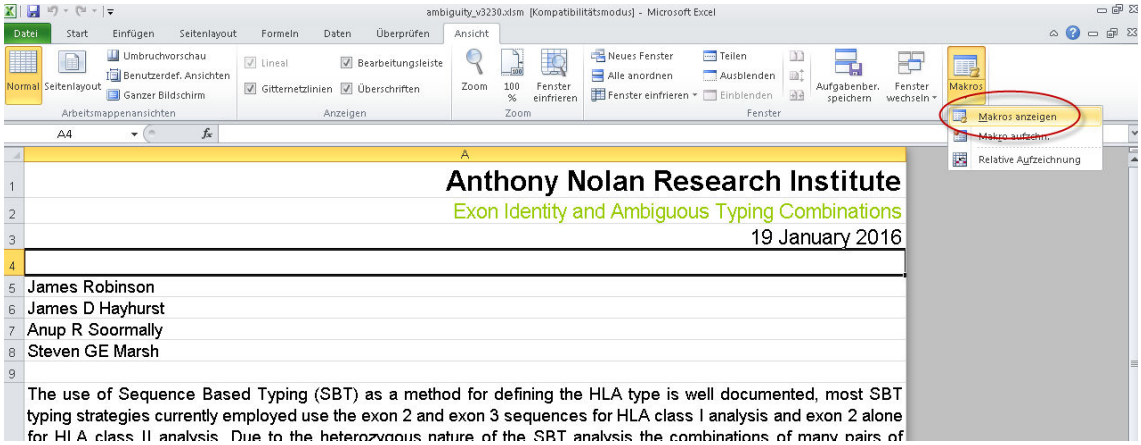
Next we extract the information from the Excel sheet. Open ambiguity_v<>.xls in Excel and save as ambiguity_v<>.xlsm to run macros.

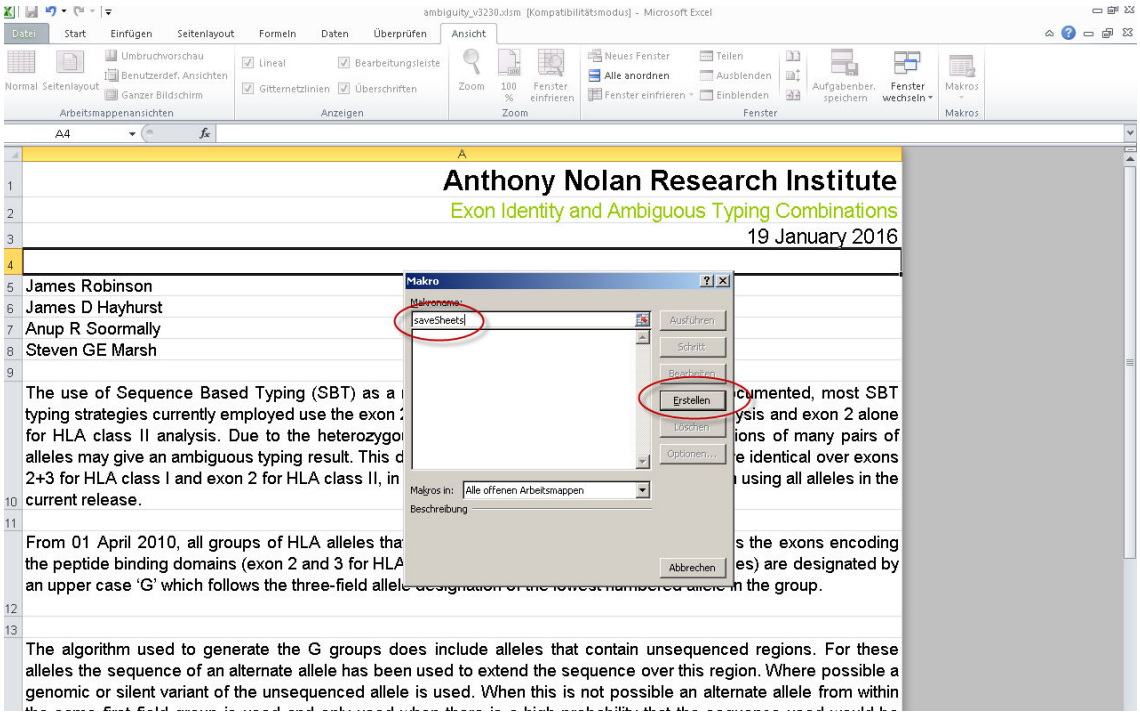


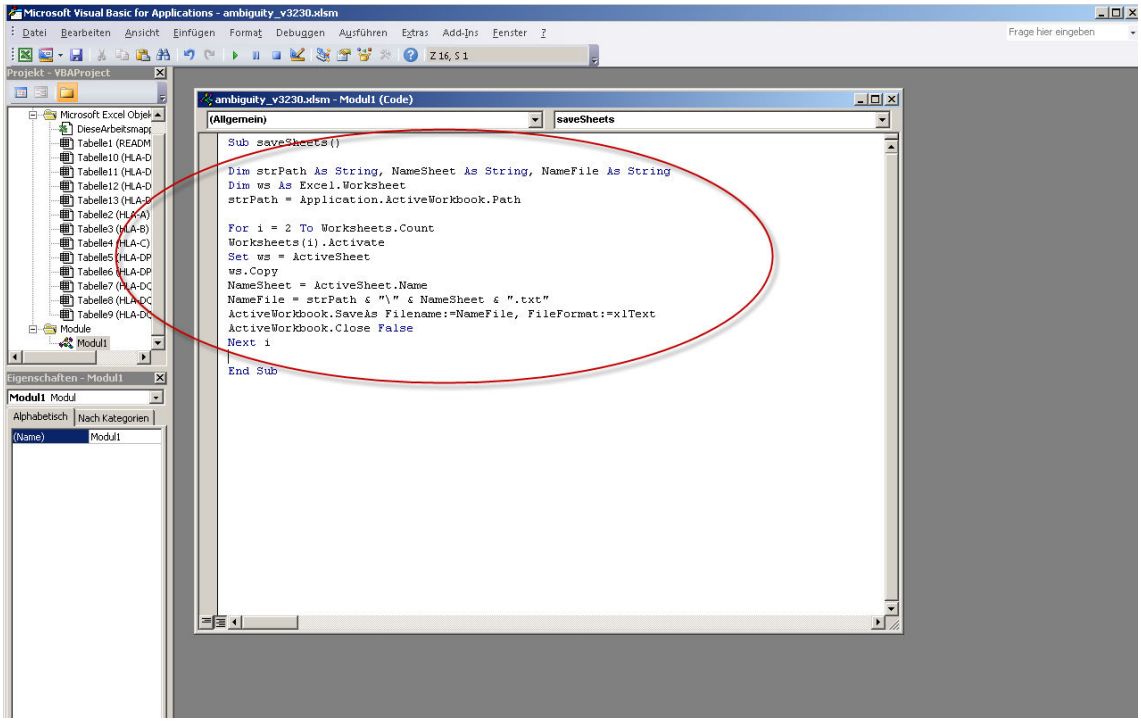


Now insert the following macro, which saves relevant information from the Excelsheets as text files:

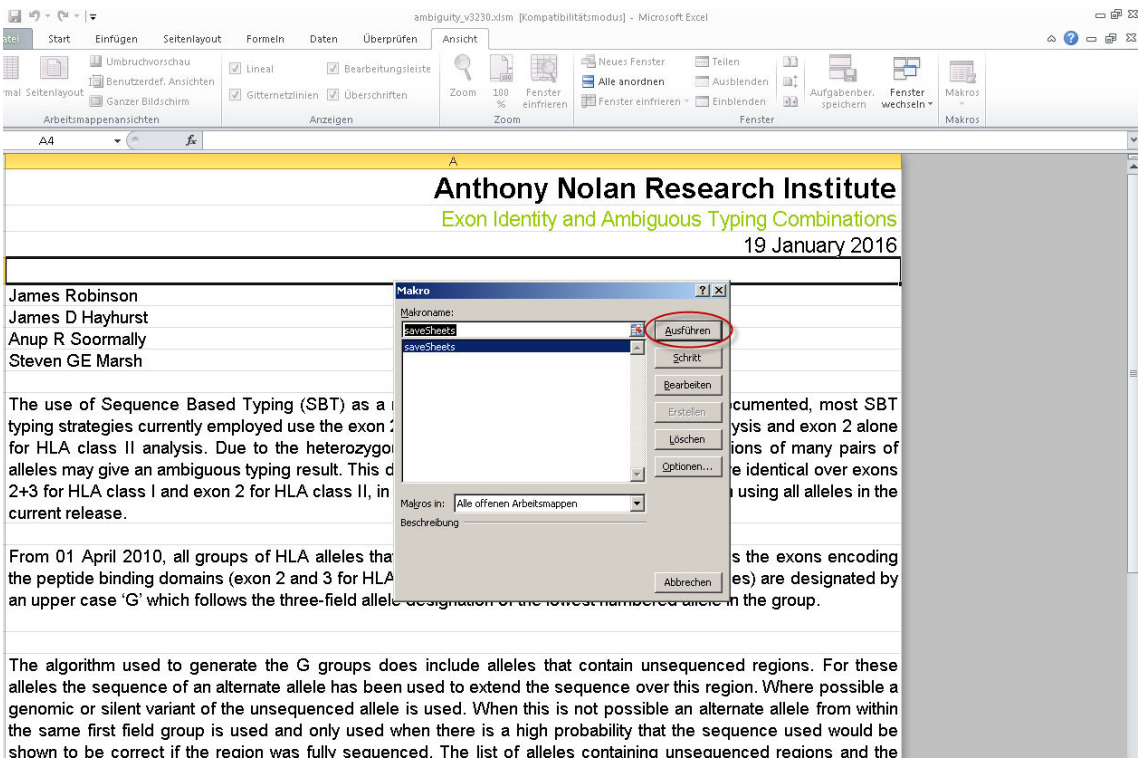
```
Sub saveSheets()
Dim strPath As String, NameSheet As String, NameFile As String
Dim ws As Excel.Worksheet
strPath = Application.ActiveWorkbook.Path

For i = 2 To Worksheets.Count
Worksheets(i).Activate
Set ws = ActiveSheet
ws.Copy
NameSheet = ActiveSheet.name
NameFile = strPath & "\" & NameSheet & ".txt"
ActiveWorkbook.SaveAs Filename:=NameFile, FileFormat:=xlText
ActiveWorkbook.Close False
Next i
End Sub
```


ambiguity_v3230.xlsm [Kompatibilitätsmodus] - Microsoft Excel
Datei
Start
Einfügen
Seitenlayout
Formeln
Daten
Überprüfen
Ansicht
Makros
Makros anzeigen
Makro aufzeichn.
Relative Aufzeichnung
A4
Anthony Nolan Research Institute
Exon Identity and Ambiguous Typing Combinations
19 January 2016
James Robinson
James D Hayhurst
Anup R Soormally
Steven GE Marsh
The use of Sequence Based Typing (SBT) as a method for defining the HLA type is well documented, most SBT typing strategies currently employed use the exon 2 and exon 3 sequences for HLA class I analysis and exon 2 alone for HLA class II analysis. Due to the heterozygous nature of the SBT analysis the combinations of many pairs of


ambiguity_v3230.xlsm [Kompatibilitätsmodus] - Microsoft Excel
Anthony Nolan Research Institute
Exon Identity and Ambiguous Typing Combinations
19 January 2016
James Robinson
James D Hayhurst
Anup R Soormally
Steven GE Marsh
Makro
Makroname:
saveSheets
Ausführen
Schritt
Bearbeiten
Erstellen
Löschen
Optionen...
Makros in:
Alle offenen Arbeitsmappen
Beschreibung
Abbrechen
current release.
From 01 April 2010, all groups of HLA alleles
the peptide binding domains (exon 2 and 3 for HLA
an upper case 'G' which follows the three-field allele
The algorithm used to generate the G groups does include alleles that contain unsequenced regions. For these alleles the sequence of an alternate allele has been used to extend the sequence over this region. Where possible a genomic or silent variant of the unsequenced allele is used. When this is not possible an alternate allele from within



Close the window and execute the macro:



A bunch of text files as “HLA-A.txt” and so on should have appeared. Move these to the folder “Hapl-o-Mat/prepareData”. Afterwards you can remove the Excel file.

Build Data for Hapl-o-Mat

Now your are ready to build the data. Enter the folder “Hapl-o-Mat/prepareData” and run the following python scripts in the given order:

1) python BuildAllAllelesFrom_hla_nom_g.py
2) python BuildAllAllelesExpanded.py
3) python BuildP.py
4) python BuildLargeG.py
5) python BuildSmallg.py
6) python AddAllelesMissingIngCode.py
7) python TransferAlphaToMultipleAlleleCodes
8) If you went in the last section with the xml-file, run python BuildAmbiguityFromXML.py , if you went with the Excel sheet, run python BuildAmbiguityFromTextFiles.py.
9) python AddGToAmbiguity.py

Next, create the folder “Hapl-o-Mat/data” and move the freshly created files LargeG.txt, P.txt, Smallg.txt, Ambiguity.txt, MultipleAlleleCodes.txt, and AllAllelesExpanded.txt there. You can remove the files alpha.v3.txt, hla_ambigs.xml, hla_nom_g.txt, hla_nom_p.txt, allAlleles.txt, and OneElementG.txt.

If you want to analyse data in GL-format with unresolved genotypes (GL-id=0), you can prepare the file AlleleList.txt from the GL-id input file by running BuildAlleleList.py. Then move AlleleList.txt to data.