Handbuch „afxStatGUI“

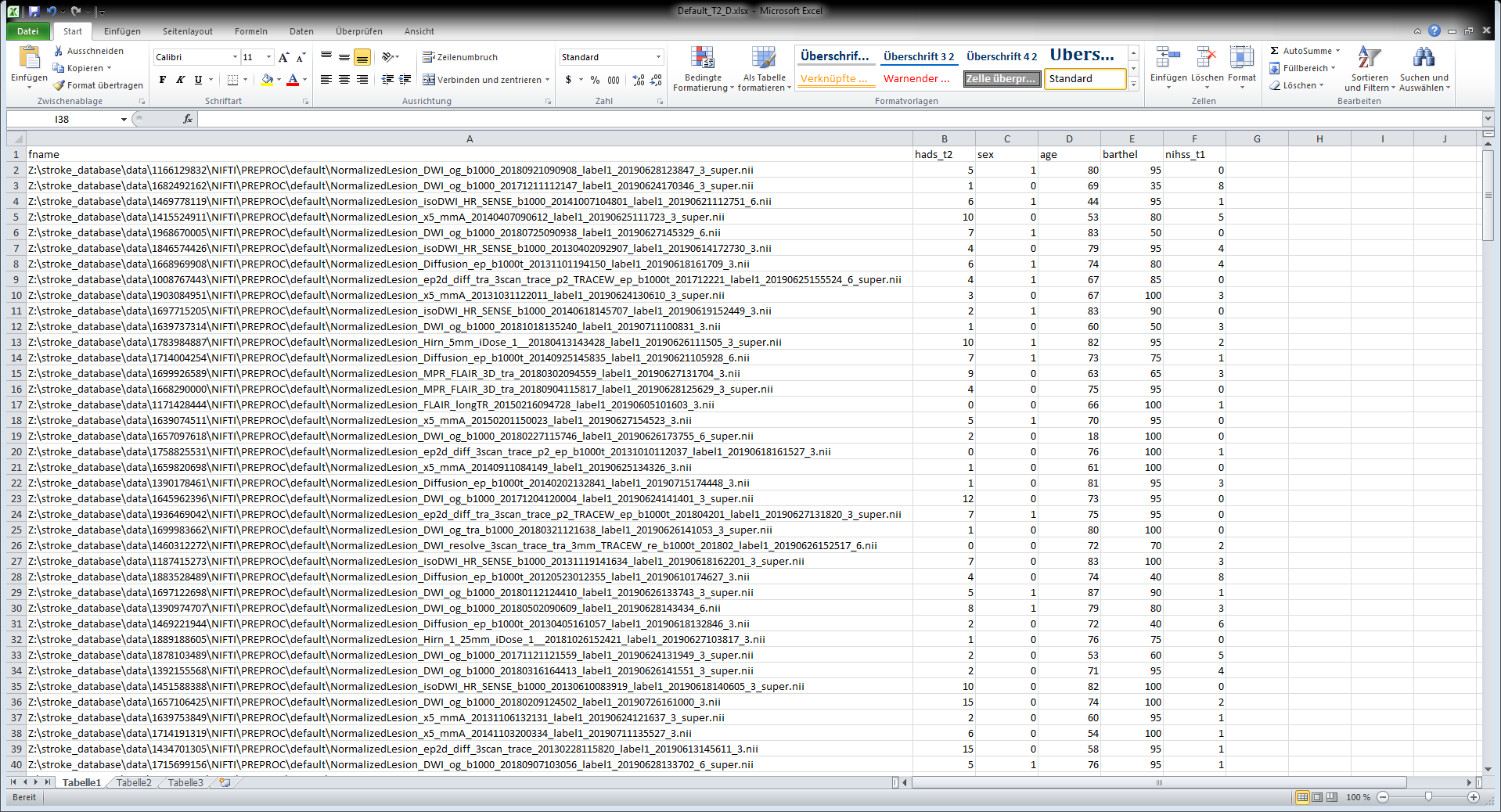
# Einleitung

AfxStatGUI dient der voxelbasierten Analyse von Läsionsdaten mittels voxel-based lesion-symptom mapping (VLSM) und lesion network-symptom mapping (LNSM). Die statistische Inferenz beruht auf linearen Modellen und zufälligen Permutationstests. Alle Verarbeitungsschritte bis zur in den Standardraum normalisierten Läsion finden innerhalb des Frameworks der StrokeDb statt (siehe dort). Die Schnittstelle zwischen StrokeDb und afxStatGUI bilden Exceltabellen.

# Vorbereitung

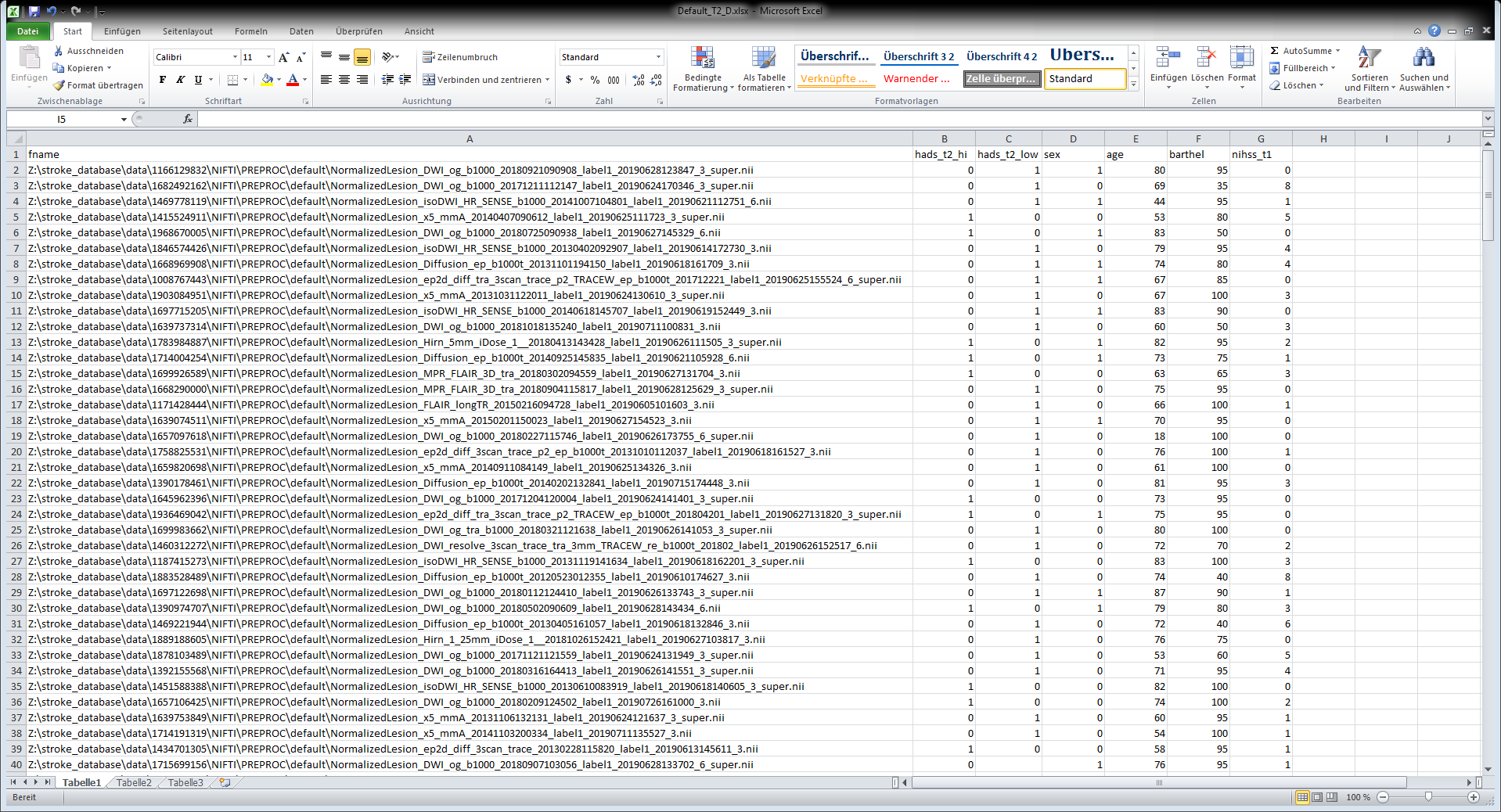
In Vorbereitung einer Läsionsanalyse steht die saubere Definition einer Designmatrix mit Excel. Grundlage hierfür bilden die aus der StrokeDb exportierten Tabellen. Grundlage für alle Analysen bilden allgemeine lineare Modelle der Form:

Dabei sind Y die zu erklärenden Daten (also die Läsionsdaten), X ist die Designmatrix, Beta sind die Modellkoeffizienten, Epsilon ist die residuelle Varianz.



**Abbildung 1**: Beispiel einer Designmatrix mit metrischen Daten (t-test).

Analog muss die erste Spalte der Exceltabelle enthält den vollständigen Dateinamen der Läsion auf dem Broca-Server (Y). (Hierzu muss der entsprechende Pfad unter Z: eingehängt sein, siehe StrokeDb-Manual.). Alle weiteren Spalten enthalten die Designmatrix (X), wobei die erste Spalte der Designmatrix den zu untersuchenden Effekt enthalten muss (Abbildung 1). Die erste Zeile enthält die Spaltenüberschriften. Regressoren (Spalten) der Designmatrix können beliebige Skalenniveaus besitzen. Bei binären Daten ist eine zweite Spalte mit umgekehrter Kodierung hinzuzufügen (Abbildung 2).



**Abbildung 2**: Beispiel einer Designmatrix mit binären Daten.

# Benutzung von afxStatGUI

## Allgemeine Benutzung am Beispiel von Voxel-based lesion-symptom mapping (VLSM)

* Matlab starten
* In Toolbox-Verzeichnis wechseln:

cd Z:\projects\strokedb\statistics\afxStat-0.9

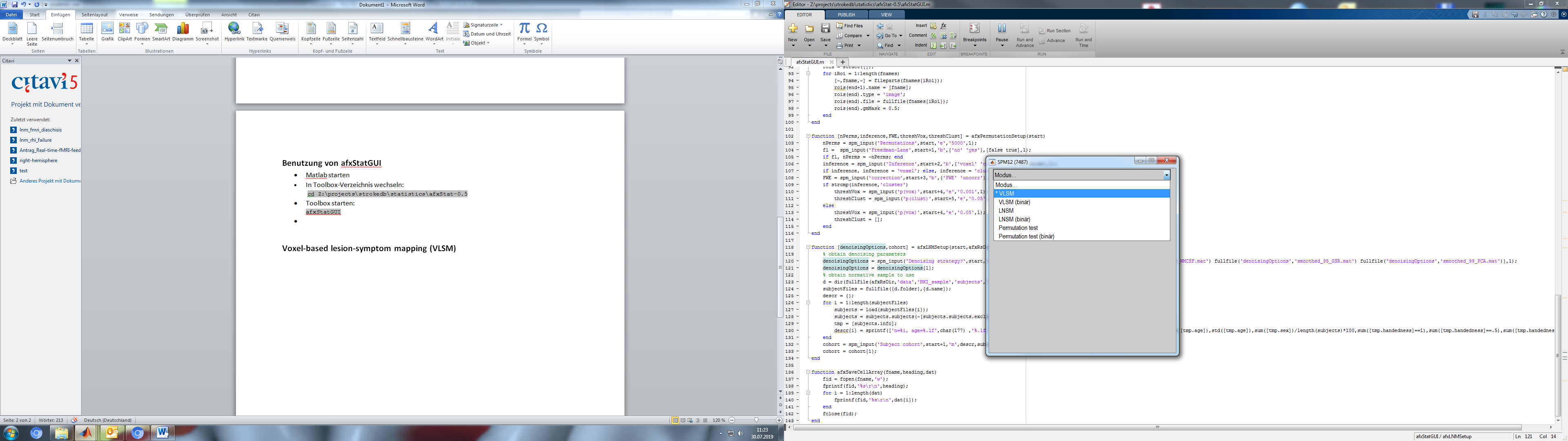
(Diesen Ordner NICHT zum Matlabpfad hinzufügen)

* Toolbox starten:

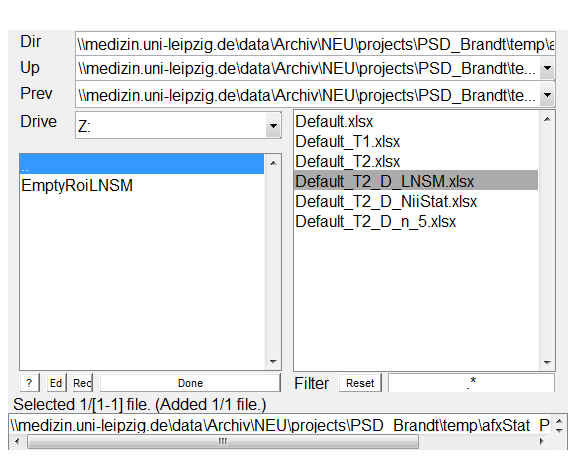
afxStatGUI

* Modus „VLSM“ (kontinuierliche Daten) oder „VLSM (binär)“ (Gruppenvergleiche) wählen:

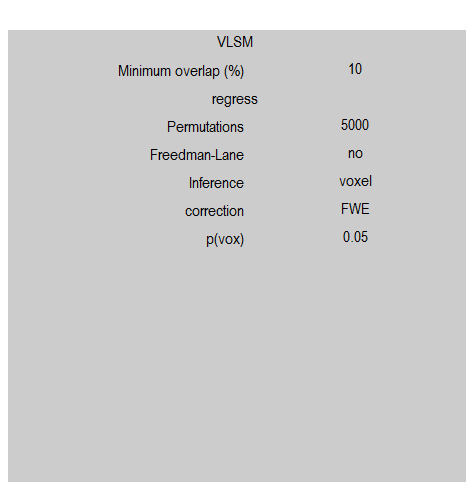
(„LNM (1st-level)“ für das Lesion network mapping, „LNSM (2nd-level)“ für Secondlevel-Analysen, wenn schon Lesion networks vorhanden sind.)



* Design file (Excel) wählen:

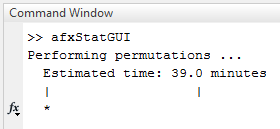


* Weitere Einstellungen vornehmen:



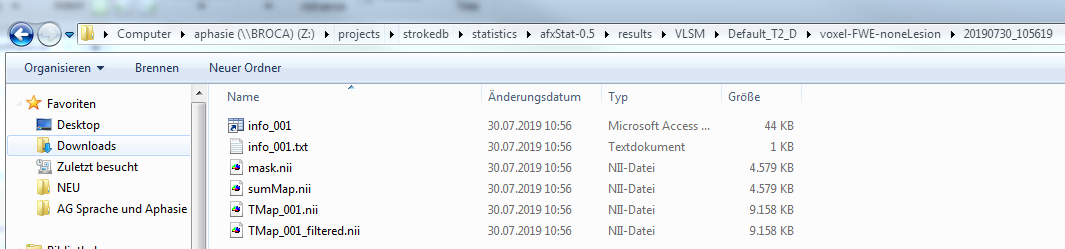
|  |  |
| --- | --- |
| **Einstellung** | **Bedeutung** |
| Minimum overlap (%) | Mindestabdeckung mit Läsionen, damit ein Voxel an der Analyse teilnehmen kann, üblicherweise 5 %.  (Für absolute Werte negative Zahlen eingeben, z.B. „-5“ für einen minimalen Overlap von 5 Läsionen, siehe z.B. Sperber et al., 2017, die einen Overlap von mindestens 5 vorschlagen.) |
| Account for lesion volume? | Verhaltensmaße sind in der Regel mit der Läsionsgröße assoziiert. Daher muss die Läsionsgröße in das Modell einfließen. |
| Permutations | Anzahl der zufälligen Permutationen. |
| Freedman-Lane | Sollte gewählt werden, wenn die Designmatrix mehr als eine Spalte enthält. |
| Inference | Statistische Inferenz auf Voxel- oder Clusterebene? (Clusterebene ist sensitiver aber für VLSM kaum verbreitet.) |
| Correction | FWE-Korrektur sollte immer gewählt werden. |
| p(vox) | Signifikanzniveau auf Voxelebene. |
| P(clust) | Signifikanzniveau auf Clusterebene. |

* Die Analyse wird nun durchgeführt. Im Command Window von Matlab wird man über den Arbeitsfortschritt informiert:



* Die Ergebnisse finden sich nach Abschluss der Analyse in folgendem Ordner:

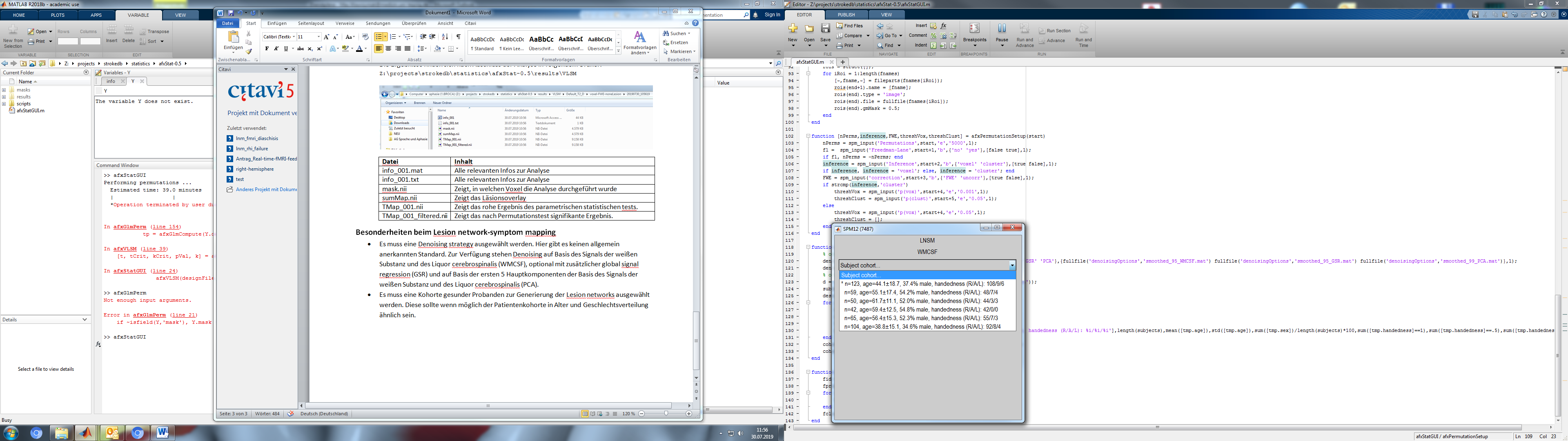
Z:\projects\strokedb\statistics\afxStat-0.5\results\VLSM



|  |  |
| --- | --- |
| **Datei** | **Inhalt** |
| info\_001.mat | Alle relevanten Infos zur Analyse |
| info\_001.txt | Alle relevanten Infos zur Analyse |
| mask.nii | Zeigt, in welchen Voxel die Analyse durchgeführt wurde |
| sumMap.nii | Zeigt das Läsionsoverlay |
| TMap\_001.nii | Zeigt das rohe Ergebnis des parametrischen statistischen tests. |
| TMap\_001\_filtered.nii | Zeigt das nach Permutationstest signifikante Ergebnis. |

## Besonderheiten beim Lesion network-symptom mapping

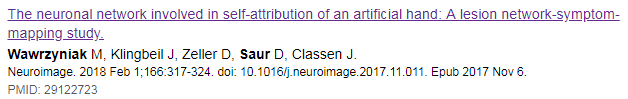
* Es muss eine Denoising strategy ausgewählt werden. Hier gibt es keinen allgemein anerkannten Standard. Zur Verfügung stehen Denoising auf Basis des Signals der weißen Substanz und des Liquor cerebrospinalis (WMCSF), optional mit zusätzlicher global signal regression (GSR) und auf Basis der ersten 5 Hauptkomponenten der Basis des Signals der weißen Substanz und des Liquor cerebrospinalis (PCA).
* Es muss eine Kohorte gesunder Probanden zur Generierung der Lesion networks ausgewählt werden. Diese sollte wenn möglich der Patientenkohorte in Alter und Geschlechtsverteilung ähnlich sein. Es stehen verschiedene Kohorten zur Verfügung (s.u.). Es ist auch möglich weitere Kohorten zu definieren, jedoch ist das mittlere Alter nur unter deutlicher Abnahme der Fallzahl möglich.



* Es werden alle Patienten ausgeschlossen, deren Läsion bei mindestens einem gesunden Probanden gänzlich im Bereich der weißen Substanz liegt. Diese Patienten sind am Ende bei den Ergebnissen in der Datei excluded\_patients.txt vermerkt.
* Der Schwellwert für das gray matter masking wird ebenfalls erfragt. Für die NKI-Daten empfiehlt sicht 0.5, für die HCP-Daten 0.2.
* Nach Berechnung der Lesion networks (zeitaufwändig) wird neben dem ursprünglichen Designfile ein weiteres mit einem längeren Dateinamen (z.B. „…\_LNSM\_subjects\_matched\_ahp\_stroke-smoothed\_95\_WMCSF“) abgelegt, welches dann für Second-level Analysen (Modus „LNSM (2nd-level)“ oder „LNSM (2nd-level, binär)“) genutzt werden kann, ohne das Lesion network mapping erneut durchführen zu müssen.
* Im 2nd-level muss eine Maske gewählt werden. Für whole-brain Analyse bietet sich „brainmask.nii“ an. Für ROI-Analysen kann eine beliebige Maske gewählt wertden.

# Nachbemerkungen

Literaturempfehlung zum LNSM:



Bei Problemen bitte an Max Wawrzyniak ([max.wawrzyniak@medizin.uni-leipzig.de](mailto:max.wawrzyniak@medizin.uni-leipzig.de)) wenden.

Leipzig, den 01.07.2020