COVID-19 in MS

Global Data Sharing Initiative

Analyses Plan 2020

APPENDIX

Table A1: Data quality assessment and enhancement pipeline - create new variables

IMPORTANT NOTE: we assume the variables are properly mapped to the format as described in the dictionary

report_source	**For patient-driven initiatives:**
	report_source = “patients”
	For clinician-driven initiatives:
	report_source = “clinicians”
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

covid19_diagnosis	**For patient-driven initiatives:** **Covid19_diagnosis= “confirmed”** if covid19_confirmed_case=”yes” **Covid19_diagnosis= “suspected”;** Includes two options (OR statement): ● [covid19_sympt_fever=yes OR covid19_sympt_dry_cough=yes OR covid19_sympt_loss_smell_taste=Yes] **AND** declared suspicion by patient (covid19_suspected_case=yes) ● [covid19_sympt_fever=yes OR covid19_sympt_dry_cough=yes OR covid19_sympt_loss_smell_taste=Yes] **AND** at least ONE other symptom common to COVID19 ■ OR covid19_sympt_sore_throat=yes ■ OR covid19_sympt_shortness_breath=yes ■ OR covid19_sympt_pneumonia ■ OR covid19_sympt_fatigue=yes ■ OR covid19_sympt_pain=Yes ■ OR covid19_sympt_nasal_congestion=Yes ■ OR covid19_sympt_chills=Yes **For clinician-driven initiatives:** **Covid19_diagnosis= “confirmed”** if covid19_confirmed_case=”yes” **Covid19_diagnosis= “suspected”;** if covid19_suspected_case=”yes”	Done for registry (registry.py - enhanced_registry_data) Done for forms (forms.py - enhanced_forms_data)

age_in_cat	age_in_cat = 0; if 0< age_years <18
	age_in_cat = 1; if 18<= age_years <=50
	age_in_cat = 2; if 50< age_years <=70
	age_in_cat = 3; if 70< age_years
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

current_or_former_smoker	current_or_former_smoker= “yes”
	if current_smoker = “yes” or “former_smoker”=yes”
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

height and weight	transform height and weight into numeric variable 100 < height < 210 30 < weight < 300 all errors are put into NaN	Done in utils.py
bmi	bmi=weight/((height(in cm)/100)* height(in cm)/100))	Done in utils.py

bmi_in_cat	bmi_in_cat= “underweight “ if 0 < bmi < 18.5
	bmi_in_cat= “normal” if 18.5 <= bmi <= 25
	bmi_in_cat= “overweight” if 25 < bmi <= 30
	bmi_in_cat= “class I obesity” if 30 < bmi <=35
	bmi_in_cat= “class II obesity” if 35 < bmi
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

bmi_in_cat2	bmi_in_cat2 = “not_overweight” if bmi <= 30
	bmi_in_cat2 = “overweight” if bmi > 30
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

year_onset	extract year from ms_onset_date	Done for registry (registry.py - enhanced_registry_data) Done for forms (forms.py - enhanced_forms_data)
year_reporting	extract year from covid19_date_reporting	Done for registry (registry.py - enhanced_registry_data) Done for forms (forms.py - enhanced_forms_data)
disease_duration	year_reporting - year_onset	Done for registry (registry.py - enhanced_registry_data) Done for forms (forms.py - enhanced_forms_data)

disease_duration_in_cat	disease_duration_in_cat= 0; if 0< disease_duration <2
	disease_duration_in_cat= 1; if 2<= disease_duratio n<=10
	disease_duration_in_cat= 2 if 10 < disease_duration
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

disease_duration_in_cat2	disease_duration_in_cat2= 0 if disease_duration_in_cat=0 or 1 (between 0 and 10 years included)
	disease_duration_in_cat2=1 if disease_duration_in_cat=2 (above 10 years)
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

ms_type2	ms_type2 = “relapsing_remitting” if ms_type= “RRMS” ms_type2= “other” if ● ms_type= “CIS” ● or ms_type = “not_sure” ● or ms_type = “” ms_type2= “progressive_MS” if ● ms_type= “SPMS” ● or ms_type = “PPMS”	Done for registry (registry.py - enhanced_registry_data) Done for forms (forms.py - enhanced_forms_data)

edss_in_cat	edss_in_cat=0
	if 0<= edss <=3
	edss_in_cat=1
	if 3< edss <=6
	edss_in_cat=2
	if 6< edss
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

edss_in_cat2	edss_in_cat2=”zero” if 0<= edss <=6
	edss_in_cat2=”one” if 6< edss
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

type_last_dmt	if (current_dmt is “no” and dmt_stop_date-covid19_date_reporting < 180 days) OR (current_dmt is “yes”)
	type_last_dmt = type_dmt
	else :
	type_last_dmt = None	

type_current_dmt	type_current_dmt = type_dmt if current_dmt = “yes”. Else, type_current_dmt = None	

dmt_type_overall	dmt_type_overall= “No information on DMT use” if current_dmt= missing or type_dmt = missing
	dmt_type_overall= “currently not using any DMT” if current_dmt= “no” OR “no, but was in the past”
	dmt_type_overal= “currently on interferon” if current_dmt = “yes” AND type_dmt = “interferons”
	dmt_type_overall= “currently on glatiramer” if current_dmt = “yes” AND type_dmt = “glatiramer”
	dmt_type_overall= “currently on natalizumab” if current_dmt = “yes” AND type_dmt = “natalizumab”
	dmt_type_overall= “currently on fingolimod” if current_dmt = “yes” AND type_dmt = “fingolimod”
	dmt_type_overall = “currently on dimethyl fumarate” if current_dmt = “yes” AND type_dmt = “dimethyl_fumarate”
	dmt_type_overall = “currently on teriflunomide” if current_dmt = “yes” AND type_dmt = “teriflunomide”
	dmt_type_overall = “currently on alemtuzumab” if current_dmt = “yes” AND type_dmt = “alemtuzumab”
	dmt_type_overal l= “currently on cladribine” if current_dmt = “yes” AND type_dmt = “cladribine”
	dmt_type_overall = “currently on siponimod” if current_dmt = “yes” AND type_dmt = “siponimod”
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

	dmt_type_overall = “currently on rituximab” if current_dmt = “yes” AND type_dmt = “rituximab”
	dmt_type_overall = “currently on ocrelizumab” if current_dmt = “yes” AND type_dmt = “ocrelizumab”
	dmt_type_overall = “currently on another drug not listed” if current_dmt “yes” AND type_dmt_other is **not** missing	

sex_binary	sex_binary = “male” if sex = “male”
	sex_binary = “female “if sex = “female”
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

covid19_outcome_ventilation_or_ICU	covid19_outcome_ventilation_or_ICU= “yes” if covid19_ventilation = “yes” OR if covid19_icu_stay = “yes”
	covid19_outcome_ventilation_or_ICU= “no” if covid19_ventilation = “no” AND if covid19_icu_stay = “no”
	else : None
		Done for registry (registry.py - enhanced_registry_data)
		Done for forms (forms.py - enhanced_forms_data)

Table A2: Data quality assessment and enhancement - PASS and FAIL criteria

IMPORTANT NOTE: we assume the variables are properly mapped to the format as described in the dictionary

Here, we decide upon PASS versus FAIL criteria for every variable as well as how we are going to cope with this in downstream analysis. In the FAIL versus PASS criteria column, the variables are flagged differently according to the data quality check*:* *pass* is the flag for an accepted variable, *fail* is the flag for a dismissed variable, *<empty>* is the flag for each variable that's missing/null. *Fail* does not mean that data gets excluded, just flagged as erroneous - it can also be adapted in some cases for analysis. Currently a 'fail' means the following action: "set failed variable to missing, flag the variable, keep the entry (row)".**Essential variables for the current analysis are marked green.** All dates cannot be in the future (anydate > date_reporting then fail) or before a persons birthday (anydateYEAR < (year_reporting - age) then 'fail'), the COVID related dates have to be >= MS baseline dates (onset, diagnosis)

Variable	format	Interdependency	FAIL versus PASS criteria	Implemented (grey means disabled in current cleaning)

covid19_date_reporting	YYYY-MM-DD	None	if covid19_date_reporting < 2019 then 'fail'
			else pass
				cleaning_functions.py - clean_covid19_date_reporting

covid19_has_symptoms	single choice (yes/no)	covid19_sympt_fever covid19_sympt_dry_cough covid19_sympt_fatigue covid19_sympt_pain covid19_sympt_sore_throat covid19_sympt_shortness_breath covid19_sympt_nasal_congestion covid19_sympt_lass_smell_taste covid19_sympt_pneumonia	if covid19_has_symptoms == null then check the covid19_sympt_xx for 'yes' if any covid19_sympt_xx = 'yes' then covid19_has_symptoms = 'yes' (for the analysis data) strict: if covid19_has_symptoms = 'no' AND any covid19_symp_xx = 'yes' then 'fail' *derivation: covid19_has_symptoms is secondary to covid19_sympt_xx - if any of the single symptoms are 'yes' the empty(!) covid19_has_symptoms will be set to 'yes' and vice versa (all symptoms = 'no', covid19_has symptoms is set to 'no')*	cleaning_functions.py - clean_covid19_has_symptoms repair covid19_has _symptoms: cleaning_functions.py - repair_covid19_has_symptoms
covid19_sympt_fever	Single choice (yes/no)	covid19_has_symptoms	see covid19_has_symptoms	
covid19_sympt_dry_cough	Single choice (yes/no)	covid19_has_symptoms	see covid19_has_symptoms	
covid19_sympt_fatigue	Single choice (yes/no)	covid19_has_symptoms	see covid19_has_symptoms	

covid19_sympt_pain	Single choice (yes/no)	covid19_has_symptoms	see covid19_has_symptoms	
covid19_sympt_sore_throat	Single choice (yes/no)	covid19_has_symptoms	see covid19_has_symptoms	
covid19_sympt_shortness_breath	Single choice (yes/no)	covid19_has_symptoms	see covid19_has_symptoms	
covid19_sympt_nasal_congestion	Single choice (yes/no)	covid19_has_symptoms	see covid19_has_symptoms	
covid19_sympt_chills	Single choice (yes/no)	covid19_has_symptoms	see covid19_has_symptoms	
covid19_sympt_loss_smell_taste	Single choice (yes/no)	covid19_has_symptoms	see covid19_has_symptoms	
covid19_sympt_pneumonia	Single choice (yes/no)	covid19_has_symptoms	see covid19_has_symptoms	
covid19_suspected_case	Single choice (yes/no)	covid19_confirmed_case		This is done in the analysis: see utils.py - create_covid19_diagnosis
covid19_self_isolation	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
covid19_self_isolation_by_self_patient	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
covid19_self_isolation_date	Date (YYYY-MM-DD)		if covid19_self_isolation_date < 2019 OR covid19_self_isolation_date > currentDate then ‘fail’	cleaning_functions.py - clean_covid19_self_isolation_date
covid19_self_isolation_duration	Number	covid19_date_reporting covid19_self_isolation_date	if covid19_self_isolation_duration > (covid19_date_reporting - covid19_self_isolation_date) then fail	cleaning_functions.py - clean_covid19_self_isolation_duration
covid19_confirmed_case	Single choice (yes/no)	covid19_suspected_case	Interdependency with covid19_suspected_case is fixed when defining covid19_diagnosis	This is done in the analysis: see

				utils.py - create_covid19_diagnosis
covid19_date_lab_test	Date (YYYY-MM-DD)		covid19_date_lab_test < 2019 OR covid19_date_lab_test > currentDate then 'fail'	cleaning_functions.py - clean_covid19_date_lab_test
covid19_country	Single choice (Country name)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
covid19_date_suspected_onset	Date (YYYY-MM-DD)		covid19_date_suspected_onset < 2019 AND > currentDate then 'fail'	cleaning_functions.py - clean_covid19_date_suspected_onset
covid19_admission_hospital	Single choice (yes/no)		if covid19_admission_hospital = 'yes' AND covid19_confirmed_case = 'no' then fail *Suggestion to sort out the hospital admissions that are unrelated to COVID*	cleaning_functions.py - clean_covid19_admission_hospital

covid19_admission_hospital_date	Date (YYYY-MM-DD)	covid19_admission_hospital_release	if covid19_admission_hospital_date > covid19_admission_hospital_release then 'fail'
			else 'pass'
				cleaning_functions.py -
				clean_covid19_admission_hospital_release

covid19_admission_hospital_release	Date (YYYY-MM-DD)	covid19_admission_hospital_date	if covid19_admission_hospital_release =! null
			else if covid19_admission_hospital_release < covid19_admission_hospital_date then 'fail'
			else 'pass'
				cleaning_functions.py -
				clean_covid19_admission_hospital_release

covid19_icu_stay	Single choice (yes/no)	covid19_admission_hospital	if covid19_icu_stay = 'yes' AND covid19_admission_hospital = 'no'
			then 'fail'
			else 'pass'
			repair:
			if covid19_icu_stay empty AND covid19_admission_hospital = no
			then covid19_icu_stay = no
				cleaning_functions.py -
				clean_covid19_icu_stay
				cleaning_functions.py -
				repair_covid19_icu

covid19_still_icu_stay	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	

covid19_icu_current_duration	Number		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
covid19_icu_total_duration	Number		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	

covid19_ventilation	Single choice (yes/no)	covid19_ventilation_non_invasive
		covid19_ventilation_invasive
			if (covid19_ventilation == null OR ‘no’) AND (covid19_ventilation_invasive = ’yes’ OR covid19_ventilation_non_invasive = ‘yes’)) OR (covid19_ventilation = ‘no’ AND (covid19_ventilation_invasive = ’yes’ OR covid19_ventilation_non_invasive = ‘yes’))
			then ‘fail’
			else ‘pass’
			repair:
			if covid19_vantilation is empty AND covid19_ventilation_invasive or/and covid19_ventilation_non_invasive = ‘yes’
			then covid19_ventilation = ‘yes’
			if covid19_ventilation is empty AND covid19_admission hospital = ‘no’
			then covid19_ventilation = ‘no’
				cleaning_functions.py
				clean_covid19_ventilation
				
				repair::
				repair_covid19_ventilation

covid19_ventilation_non_invasive	Single choice (yes/no)		see covid19_ventilation	
covid19_ventilation_invasive	Single choice (yes/no)		see covid19_ventilation	
covid19_ecmo	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
covid19_outcome_recovered	Single choice (yes/no/not_applicable)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	

covid19_outcome_death	Single choice (yes/no)	covid19_outcome_recovered	if covid19_outcome_death = ‘yes’ AND covid19_outcome_recovered = ‘yes’ then ‘fail’
			else pass
			
			repair:
			if covid19_outcome_death is empty AND covid19_outcome_recovered = ‘yes’ then covid19_outcome_death = ‘no’
				cleaning_functions.py - clean_covid19_outcome_death
				
				cleaning_functions.py - repair_covid19_outcome_death

covid19_outcome_death_date	Date (YYYY-MM-DD)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	

age_years	Number		if age_years < 0 OR age_years >110 then fail
			else pass
				cleaning_functions.py - clean_age_years

sex	Single choice (male/female/ non-binary)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	

pregnancy	Single choice (yes/no)	sex	if pregnancy = ‘yes’ AND sex = ‘male’ then fail
			else pass
				cleaning_functions.py - clean_pregnancy

current_smoker	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	

former_smoker	Single choice (yes/no)	current_smoker	if former_smoker = ‘yes’ AND current_smoker = ‘yes’ then ‘fail’
			else pass
				cleaning_functions.py - clean_former_smoker

height	Number		if height > 210 or height <100 then fail	cleaning_functions.py - clean_height
weight	Number		if weight> 300 or weight<30 then fail	cleaning_functions.py - clean_weight
is_healthcare_profession	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
ms_type	Single choice (CIS, RRMS, SPMS, PPMS, not_sure)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	

ms_onset_date	Date (YYYY-MM-DD)	ms_diagnosis_date
		covid19_suspected_onset
			if (ms_onset_date > ms_diagnosis_date) OR (ms_onset_date > covid19_suspected_onset) then ‘fail’
			else pass
				cleaning_functions.py - clean_ms_onset_date

ms_diagnosis_date	Date (YYYY-MM-DD)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	

edss_date_diagnosis	Date (YYYY-MM-DD)		if year of edss_date < 2019 then fail *to sort out older EDSS values that are in no direct relation to a covid 19 infection*	cleaning_functions.py - clean_edss_date_diagnosis
edss_value	Number (0.0, 10.0)		if edss_value < 0 OR edss_value > 10 then 'fail'	cleaning_functions.py - clean_edss_value
last_white_blood_cell	Number		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
last_white_blood_cell_unit	Text		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
last_lympho_cell	Number		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
last_lympho_cell_unit	Text		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
last_b_cell	Number		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
last_b_cell_unit	Text		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
current_dmt	Single choice (yes/no/never_treated)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	

type_dmt	Single choice
	(interferons/glatiramer/natalizumab/fingomilod/teriflunomide/alemtuzumab/ocrelizumab/cladribine/siponimod/rituximab/dimethyl_fumarate)
			if type_dmt == null AND type_dmt_other == null AND current_dmt = 'yes'
			then fail
			else pass
				cleaning_functins.py - clean_type_dmt

type_dmt_other	Text		if type_dmt_other is empty
			type_dmt_other = None
				cleaning_functins.py - repair_type_dmt_other

dmt_start_date	Date (YYYY-MM-DD)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
dmt_end_date	Date (YYYY-MM-DD)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
dmt_stop_date	Date (YYYY-MM-DD)		if current_dmt = no AND dmt_stop_date < 2019 then fail *to sort out older treatments that are in no direct relation to a covid 19 infection*	cleaning_functions.py - clean_dmt_stop_date
dmt_stop_reason	Multiple choice (adverse_event, pregnancy, lack_efficacy, patient_decision, onset_covid)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
dmt_glucocorticoid	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
dmt_glucocorticoid_start_date	Date (YYYY-MM-DD)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
dmt_glucocorticoid_stop_date	Date (YYYY-MM-DD)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
dmt_glucocorticoid_dosage_value	Number		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
dmt_glucocorticoid_dosage_unit	Text		No pre-processing for now, but perform feasibility study and profiling (see section below)	
has_comorbidities	Single choice (yes/no)		if has_comorbidities == null AND any com_xx = ‘yes then set has_comorbidities = ‘yes’ (for analysis)	cleaning_function.py - repair_has_comorbidities

com_cardiovascular_disease	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
com_hypertension	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
com_diabetes	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
com_chronic_liver_disease	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
com_chronic_kidney_disease	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
com_neurological_neuromuscular	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
com_lung_disease	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
com_immunodeficiency	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
com_malignancy	Single choice (yes/no)		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	
com_other	Text		No pre-processing for now, but feasibility study and profiling is performed on the data in the central platform.	