TP6 : Analyse et prédiction des infections COVID-19 Nettoyage des données Importer les biblio nécessaire import numpy as np import pandas as pd Importer le fichier des données dat = pd.read csv("latestdata.csv") print('Les noms des colonnes: \n', dat.columns.values) C:\ProgramData\Anaconda3\lib\site-packages\IPython\core\interactiveshell.py:3058: DtypeWarning: Columns (1, 2,9,10,12,13,14,15,16,17,19,20,21,22,23,24,25,26,27,31,32) have mixed types. Specify dtype option on import or set low memory=False. interactivity=interactivity, compiler=compiler, result=result) Les noms des colonnes: ['ID' 'age' 'sex' 'city' 'province' 'country' 'latitude' 'longitude' 'geo_resolution' 'date_onset_symptoms' 'date_admission_hospital' 'date_confirmation' 'symptoms' 'lives_in_Wuhan' 'travel_history_dates' 'travel history location' 'reported market exposure' 'additional information' 'chronic disease binary' 'chronic disease' 'source' 'sequence available' 'outcome' 'date death or discharge' 'notes for discussion' 'location' 'admin3' 'admin2' 'admin1' 'country new' 'admin_id' 'data_moderator_initials' 'travel history binary'] In [4]: dat Out[4]: ID age sex city province country latitude longitude geo_resolution date_onset_symptoms ... date_death 000-Hong NaN male Shek Lei China 22.365019 114.133808 NaN point 1-1 Kong 000-Veneto 45.297748 11.658382 NaN male point 1-10 Euganeo 000female NaN Singapore 1.353460 103.815100 admin0 NaN 1-100 000-Zhengzhou NaN NaN Henan 34.629310 113.468000 admin2 NaN City 1000 000-Pingxiang 1-NaN NaN Jiangxi 27.513560 113.902900 admin2 NaN 10000 Coronel 2676306 female Calleria -8.378190 -74.539700 NaN point 99995 Portillo Pueblo 2676307 female Lima -12.076530 -77.067350 NaN point 99996 female Comas Lima Peru -11.932980 -77.040850 point NaN 99997 2676309 male Callao Peru -12.000740 -77.118240 point NaN ... 2676310 male Los Olivos Lima Peru -11.976260 -77.075030 point NaN ... 2676311 rows × 33 columns crée un sous-ensemble de data df = dat.loc[:, ['ID', 'age', 'sex', 'symptoms', 'chronic_disease','chronic_disease_binary', 'outcome']] #df.to_csv('covid_subset.csv', index=False) #print('The names of all the columns after subseting in the data: \n',df.columns.values) Out[6]: outcome ID symptoms chronic_disease chronic_disease_binary age 0 000-1-1 NaN False critical condition, intubated as of 14.02.2020 NaN NaN male 000-1-10 78 male NaN NaN False death 2 000-1-100 61 female NaN NaN False discharge 000-1-1000 NaN NaN NaN False NaN NaN 4 000-1-10000 NaN NaN False NaN NaN NaN 010-99995 52 female 2676306 NaN NaN False NaN 2676307 010-99996 52 female NaN NaN False NaN 2676308 010-99997 52 female NaN NaN False NaN 2676309 010-99998 False male NaN NaN NaN 52 2676310 010-99999 NaN False NaN 52 male NaN 2676311 rows × 7 columns Combien de null df.isnull().sum(axis = 0)Out[7]: ID 0 2098293 2096154 2674259 symptoms chronic disease 2676096 chronic_disease_binary 0 outcome 2368929 dtype: int64 Outcome In [8]: df['outcome'].value counts() Out[8]: Hospitalized 202475 93656 Recovered Deceased 5011 recovered 3659 916 370 Under treatment Receiving Treatment 257 Alive discharge 175 stable 156 stable condition 133 https://www.mspbs.gov.py/covid-19.php discharged 117 death 48 Stable 31 Dead 13 Migrated Other Migrated 5 Discharged from hospital 4 released from quarantine Died Death dead severe not hospitalized recovering at home 03.03.2020 Symptoms only improved with cough. Currently hospitalized for follow-up. critical condition, intubated as of 14.02.2020 treated in an intensive care unit (14.02.2020) Critical condition 1 critical condition unstable 1 severe illness 1 Name: outcome, dtype: int64 df['outcome'].fillna('discharged', inplace=True) df['outcome'].value counts() Out[10]: discharged 2369046 Hospitalized 202475 93656 Recovered Deceased 5011 3659 recovered died 916 Under treatment 370 Receiving Treatment 257 189 Alive discharge 175 156 stable stable condition 133 126 https://www.mspbs.gov.py/covid-19.php 48 death Stable 31 Dead 13 Migrated Other 8 Migrated Discharged from hospital released from quarantine Died 4 Death 4 dead severe not hospitalized recovering at home 03.03.2020 Symptoms only improved with cough. Currently hospitalized for follow-up. critical condition, intubated as of 14.02.2020 Discharged treated in an intensive care unit (14.02.2020) Critical condition critical condition unstable severe illness Name: outcome, dtype: int64 SEX df.isnull().sum(axis = 0)Out[11]: ID 0 2098293 2096154 symptoms 2674259 chronic disease chronic_disease_binary outcome dtype: int64 df['sex'].value_counts() Out[12]: male 307188 female 272969 Name: sex, dtype: int64 df['sex'].fillna('male', inplace=True) # car le mode c'est 'male' **AGE** In [14]: df = df.dropna(subset=['age']) df.shape Out[15]: (578018, 7) Symptoms & chronic disese df = df.dropna(subset=['symptoms']) In [18]: df['chronic_disease'].fillna('UNKNOWN', inplace=True) Affichage des données après remplissage des valeurs NAN df.shape ID chronic_disease chronic_disease_binary outcome age sex symptoms 476 000-1-10426 60 male fever, severe pneumonia UNKNOWN False discharged UNKNOWN 504 000-1-10451 40-49 discharged False 756 000-1-10679 20-29 cough, fever, sore throat UNKNOWN discharged male UNKNOWN 765 000-1-10687 40-49 cough, fever discharged 767 000-1-10689 80-89 dyspnea, fever False discharged UNKNOWN male 005-47758 60-69 658596 UNKNOWN False discharged male cough 671470 UNKNOWN 005-7136 40-49 male discharged 672418 005-7990 cough, fever, nausea hypertriglyceridemia True discharged 35 male UNKNOWN 672428 005-8 60-69 discharged male 674650 39 female **UNKNOWN** False discharged 006-1 mild 1577 rows × 7 columns Extraire les informations des variables "symptoms" et "chronic_disease" et les convertir en colones df['symptoms'] = df['symptoms'].str.replace(':', ',') df['symptoms'] = df['symptoms'].str.replace(';', ',') df['symptoms'] = df['symptoms'].str.replace(', ', ',') df['symptoms'] = df['symptoms'].str.lower() df['symptoms'] = df['symptoms'].str.replace('acute respiratory disease syndrome', 'respiratory problems') df['symptoms'] = df['symptoms'].str.replace('acute respiratory distress syndrome', 'respiratory problems') df['symptoms'] = df['symptoms'].str.replace('acute respiratory disease', 'respiratory problems') df['symptoms'] = df['symptoms'].str.replace('acute respiratory distress', 'respiratory problems') df['symptoms'] = df['symptoms'].str.replace('acute respiratory failure', 'respiratory problems') df['symptoms'] = df['symptoms'].str.replace('respiratory stress', 'respiratory problems') df['symptoms'] = df['symptoms'].str.replace('respiratory symptoms', 'respiratory problems') df['symptoms'] = df['symptoms'].str.replace('severe acute respiratory infection', 'respiratory problems') df['symptoms'] = df['symptoms'].str.replace('severe pneumonia', 'pneumonia') df['symptoms'] = df['symptoms'].str.replace('dry cough', 'cough') chronic_disease chronic_disease_binary ID symptoms outcome age sex UNKNOWN 476 000-1-10426 False discharged 60 fever,pneumonia male 504 000-1-10451 40-49 False discharged **UNKNOWN** male fever male cough,fever,sore throat UNKNOWN 756 000-1-10679 20-29 False discharged 765 000-1-10687 40-49 cough,fever UNKNOWN False discharged male 767 000-1-10689 80-89 UNKNOWN male dyspnea,fever False discharged 005-47758 60-69 UNKNOWN False discharged 658596 male cough UNKNOWN 671470 005-7136 40-49 mild False discharged male True discharged 672418 005-7990 35 cough,fever,nausea hypertriglyceridemia male 672428 005-8 60-69 **UNKNOWN** False discharged cough male UNKNOWN 674650 006-1 39 female False discharged mild 1577 rows \times 7 columns sym = df['symptoms'].str.get dummies(sep=',') t = sym.sum()38 ° c fever fever (38 °C) 37.1 ° c 38.7 degrees fever abdominal pain vomiting. weak weakness wheezing Length: 281, dtype: int64 In [24]: df['chronic disease'] = df['chronic disease'].str.replace(':', ',') df['chronic disease'] = df['chronic disease'].str.replace(';', df['chronic disease'] = df['chronic disease'].str.replace(', ', ',') df['chronic disease'] = df['chronic disease'].str.lower() df['chronic disease'] = df['chronic disease'].str.replace(' for more than 20 years', ',') df['chronic disease'] = df['chronic disease'].str.replace(' surgery four years ago', ',') df['chronic disease'] = df['chronic disease'].str.replace(' accident ', ',') df['chronic disease'] = df['chronic disease'].str.replace(' for five years', ',') df['chronic disease'] = df['chronic disease'].str.replace('hypertensive', 'hypertension') df['chronic disease'] = df['chronic disease'].str.replace('hypertenstion', 'hypertension') cd = df['chronic disease'].str.get dummies(sep=',') v = cd.sum()Out[24]: "thought to have had other pre-existing conditions" asthma atherosclerosis atrial fibrillation benign prostatic hyperplasia benign prostatic hypertrophy bronchial asthma cardiac disease cardiac dysrhythmia cardiomyopathy cardiovascular disease cerebral infarction cerebrovascular cerebrovascular infarct chronic bronchitis chronic kidney disease chronic obstructive pulmonary disease chronic renal insufficiency 1 colon cancer coronary artery disease coronary artery stenting coronary bypass surgery for 2 years 4 coronary heart disease coronary stenting 1 61 diabetes dislipidemia 1 dyslipidemia encephalomalacia frequent ventricular premature beat (fvpb) hemorrhage of digestive tract hepatitis b hip replacement 79 hypertension hypertension for 9 years 1 hyperthyroidism hypertriglyceridemia hypothyroidism impaired fasting glucose infarct ischemic heart disease kidney dz myeloma myocarditis on dialysis parkinson's disease pre-renal azotemia prostate cancer prostate hypertrophy renal disease stenocardia 1 taking medicine of madopar tongue cancer 1 tuberculosis type 2 diabetes for 30+ years 1455 unspecified underlying medical condition upper git bleeding valvular heart disease dtype: int64 Choisir les variables qui on plus de 40 valeurs et les convertir en colones sym = sym.loc[:, sym.sum() >= 40]cd = cd.loc[:, cd.sum() >= 40]df = pd.concat([df, sym], axis=1) df = pd.concat([df, cd], axis=1) df ID age symptoms chronic_disease chronic_disease_binary outcome asymptomatic cough fatigue fever sex 000-0 1 476 1-60 male fever,pneumonia unknown False discharged 10426 000-40-0 0 504 1male fever unknown False discharged 49 10451 000-20cough,fever,sore 756 1male unknown False discharged 29 throat 10679 000-40-0 765 1male cough,fever unknown False discharged 49 10687 000-80unknown 0 0 0 1male dyspnea,fever False discharged 89 10689 005-60-658596 0 1 0 0 male cough unknown False discharged 47758 005-40-0 0 0 671470 mild unknown discharged male 49 7136 005-1 672418 male cough,fever,nausea hypertriglyceridemia discharged 0 35 7990 60-005-8 discharged 0 672428 False 0 male cough unknown 1 69 674650 006-1 0 0 0 39 female mild unknown False discharged 1577 rows × 20 columns Convertir tous les variable en deux variables Dead:0 ou Dicharged:1 df['outcome'] = df['outcome'].str.lower() df.outcome = np.where(df.outcome.isin(['death', 'dead', 'died', 'deceased']), 0, 1 df['outcome'].value_counts() Name: outcome, dtype: int64 Convertir tous les variable en deux variables female:0 ou male:1 df.sex = np.where(df.sex == 'male', 1, 0 df['sex'].value counts() 901 Out[31]: 1 676 Name: sex, dtype: int64 symptoms chronic_disease chronic_disease_binary outcome asymptomatic cough fatigue fever hea ID age sex 000-1unknown False 0 60 fever,pneumonia 10426 000-40unknown False 0 0 1-1 fever 49 000-20cough,fever,sore 756 False 0 1unknown 29 throat 10679 000-40-1 unknown 0 cough,fever False 10687 000-80unknown False 0 0 1dyspnea,fever 10689 658596 False 0 0 cough unknown 47758 40-671470 unknown 0 mild False 7136 49 005-672418 35 1 cough,fever,nausea hypertriglyceridemia True 7990 672428 005-8 cough unknown False 674650 006-1 mild unknown False 0 0 0 1577 rows × 20 columns df.columns Out[33]: Index(['ID', 'age', 'sex', 'symptoms', 'chronic disease', 'chronic disease binary', 'outcome', 'asymptomatic', 'cough', 'fatigue', 'fever', 'headache', 'malaise', 'pneumonia', 'respiratory problems', 'runny nose', 'sore throat', 'diabetes', 'hypertension', 'unknown'], dtype='object') In [34]: df.drop(['ID', 'symptoms', 'chronic disease', 'unknown'], axis=1, inplace=True) df respiratory runn sex chronic_disease_binary outcome asymptomatic cough fatigue fever headache malaise pneumonia age problems 476 60 1 False 1 0 0 0 0 0 1 0 40-504 1 False 1 0 0 0 0 0 0 0 20-1 0 0 1 0 0 0 0 756 1 False 1 40-765 False 1 0 0 0 0 0 0 49 80-0 0 0 0 1 False 1 0 0 0 767 89 60-0 0 0 0 0 658596 1 False 1 0 1 0 40-671470 False 1 0 0 0 0 0 0 1 49 672418 35 1 True 1 0 1 0 1 0 0 0 0 60-1 0 0 0 0 0 0 0 672428 False 1 1 674650 0 0 0 0 0 0 0 39 0 False 1 0 1577 rows × 16 columns df['chronic_disease_binary'].value_counts() False 1451 126 Name: chronic disease binary, dtype: int64 df.chronic_disease_binary = np.where(df.chronic_disease_binary == True, 0 df['chronic_disease_binary'].value_counts() Out[38]: 126 Name: chronic_disease_binary, dtype: int64 df respiratory sex chronic_disease_binary outcome asymptomatic cough fatigue fever headache malaise pneumonia age problems nos 476 1 0 1 0 0 0 0 0 1 0 60 40-504 0 1 0 0 0 0 0 0 0 49 20-0 0 0 0 0 756 1 1 0 1 0 40-0 0 0 0 0 0 765 80-0 0 0 0 0 0 0 767 1 1 0 0 0 658596 0 0 0 0 0 0 671470 672418 0 0 0 1 1 60-0 0 672428 69 0 0 674650 0 0 1 0 0 0 0 0 39 1577 rows × 16 columns In [40]: df.dtypes object Out[40]: age int32 chronic disease binary int32 outcome int32 asymptomatic int64 cough int64 fatique int64 fever int64 headache int64 int64 malaise pneumonia respiratory problems int64 runny nose sore throat int64 int64 diabetes hypertension int64 dtype: object Nettoyage du variable AGE In [41]: all_age = df['age'].value_counts().index.tolist() In [42]: all age ['50-59', Out[42]: '60-69', '40-49', '70-79', '30-39', '80-89', '20-29', '33', '60', '42', '46', '55', '39', '51', '37', '36', '28', '38', '43', '35'**,** '47', '65', '52', '64', '44', '0-10', '61', '72'**,** 1291, '62', '69', '32', '56', '50', 1591, '73', '40', '80', '54', '25', '27', '68', '31', '53', '34', '57', '67', '70', 1581, '71', '45', '48', '20', 42.0, '24', '16', '22', '63', '78', '21', '26', **'**75**'**, '74', '19', 190-991, '76', 121, 1821, '30', '23', 27.0, 1861, '79', '0.25', '18-60', 56.0, '13-19', 1891, '60-60', 38.0, 1831, '81', 52.0, 69.0, 50.0, '50-69', 32.0, 37.0, '11', '88', 40.0, 191, '17', 33.0, '87', 28.0, '1', '65-', '77', 45.0, 10-61, '1.75', '10', 61.0, 72.0, 1961, 44.0, '14', 171, '18', 60.0, 161, 62.0, '15-88', 68.0, 41.0, '13', 18.0, 76.0, 31.0, '94', 64.0, '84', 13.0, 47.0, 70.0, 30.0, 66.0, 57.0, 22.0, 35.0, **'**30-35**'**] In [43]: df.age = df.age.str.replace('90-99', '95') df.age = df.age.str.replace('80-89', '85') df.age = df.age.str.replace('80-', '80') df.age = df.age.str.replace('70-79', '75') df.age = df.age.str.replace('65-', '65') df.age = df.age.str.replace('60-69', '65') df.age = df.age.str.replace('60-60', '60') df.age = df.age.str.replace('50-69', '60') df.age = df.age.str.replace('50-59', '55') df.age = df.age.str.replace('40-49', '45') df.age = df.age.str.replace('15-88', '45') df.age = df.age.str.replace('18-60', '39') df.age = df.age.str.replace('30-39', '35') df.age = df.age.str.replace('30-35', '33') df.age = df.age.str.replace('28-35', '31') df.age = df.age.str.replace('20-29', '25') df.age = df.age.str.replace('13-19', '16') df.age = df.age.str.replace('0-10', '5') df.age = df.age.str.replace('0-6', '3') df.age = df.age.astype('float') In [44]: df.isnull().sum(axis = 0)Out[44]: age 64 0 sex chronic disease binary outcome asymptomatic cough fatigue headache malaise 0 pneumonia respiratory problems runny nose sore throat diabetes hypertension dtype: int64 Calculer la moyenne des ages et remplacer NAN par cette valeur In [45]: df['age'].mean() Out[45]: 50.27263714474554 In [46]: df['age'].fillna(50, inplace=True)

Loading [MathJax]/extensions/Safe.js

