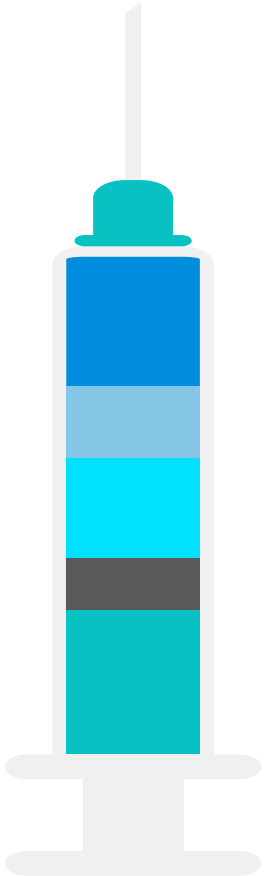
Predicting Covid Positive After Vaccination

PIPELINE

1. Ask
2. Acquire
3. Process
4. Model
5. Deliver



1. Ask: Research Question

Can we successfully build a model that can predict whether or not someone will get Covid-19 given that they have received a vaccination and if so, what model performs the best?

1. Ask: Hypothesis

Random forest classifier would provide the most accurate estimation for this classification problem since it's robust to outliers.

2. Acquire Data

Data: CDC's Vaccine Adverse Event Reporting System

1. Data Variables
   - **id**: Unique Patient ID
   - **manu**: makers of vaccine products
   - **dose**: the number of vaccine doses given for a vaccine
   - **route**: route of administration/ the path the vaccine was taken into the body
   - **site**: the site for a vaccine product
   - **recovd_date:** indicates when the patient recovered from adverse symptoms
   - **state**: Patient's Home State
   - **age**: Patient's Age
   - **sex**: Patient's Sex
   - **hospdays**: Days spent in the hospital recovering from adverse symptoms
   - **disable**: Whether or not patient is disabled
   - **recvd**: Whether or not patient recovered from adverse symptoms
   - **numdays**: Number of days patient had adverse symptoms
   - **adminby**: public, private, other, military, work, pharmacy, senior living, school, unknown agency
   - **hosp_visit**: Whether or not patient had to visit the hospital from adverse symptoms
   - **er_visit**: Whether or not patient had to visit the ER from adverse symptoms
   - **history**: Patient's istory of diseases
   - **allergies**: Patient's allergies
   - **other_meds**: Patient's current medications
   - **symptom1** 1st adverse symptom from vaccine
   - **symptom2**: 2nd adverse symptom from vaccine
   - **symptom3**: 3rd adverse symptom from vaccine
   - **symptom4**: 4th adverse symptom from vaccine
   - **symptom5**: 5th adverse symptom from vaccine

2. Acquire Data

	id	manu	dose	route	state	age	sex	hospdays	disable	recvd	...	other_meds	history	hosp_visit	er_visit	allergies	symptom1	symptom2	symptom3	symptom4	symptom5
0	916600	MODERNA	1	IM	TX	33.0	F	NaN	NaN	Y	...	None	None	Y	NaN	Pcn and bee venom	Dysphagia	Epiglottitis	NaN	NaN	NaN
1	916601	MODERNA	1	IM	CA	73.0	F	NaN	NaN	Y	...	Patient residing at nursing facility. See pati...	Patient residing at nursing facility. See pati...	Y	NaN	"Dairy"	Anxiety	Dyspnoea	NaN	NaN	NaN
2	916602	PFIZER\BIONTECH	1	IM	WA	23.0	F	NaN	NaN	U	...	None	None	NaN	Y	Shellfish	Chest discomfort	Dysphagia	Pain in extremity	Visual impairment	NaN
3	916604	MODERNA	1	IM	TX	47.0	F	NaN	NaN	N	...	Na	NaN	NaN	NaN	Na	Injection site erythema	Injection site pruritus	Injection site swelling	Injection site warmth	NaN
4	916606	MODERNA	1	IM	NV	44.0	F	NaN	NaN	Y	...	NaN	NaN	NaN	NaN	iodine (shellfish) has epipen	Pharyngeal swelling	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
18107	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	NaN	N	...	Meds prior to admission: Amiodarone, ASA, Lipi...	Arthritis, Afib, chronic anticoagulation, chro...	NaN	NaN	Pneumococcal vaccine - rash	Malaise	Nausea	Oxygen therapy	Parosmia	Productive cough
18108	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	NaN	N	...	Meds prior to admission: Amiodarone, ASA, Lipi...	Arthritis, Afib, chronic anticoagulation, chro...	NaN	NaN	Pneumococcal vaccine - rash	SARS-CoV-2 test positive	Streptococcus test negative	Taste disorder	Vomiting	NaN
18109	1057281	MODERNA	1	IM	MD	77.0	F	NaN	NaN	N	...	NaN	NaN	NaN	NaN	NaN	Death	NaN	NaN	NaN	NaN
18110	1057348	PFIZER\BIONTECH	1	IM	TX	88.0	F	NaN	NaN	N	...	lasik, blood pressure	congestive heart failure	Y	NaN	none	Death	Dysarthria	Dysstasia	Fatigue	Feeding disorder
18111	1057795	MODERNA	1	IM	OH	82.0	F	NaN	NaN	N	...	NaN	NaN	NaN	NaN	NaN	Death	NaN	NaN	NaN	NaN

18112 rows × 23 columns

3. Process Data

1. **Fill Specific NA Columns with Appropriate Values**
   - hospdays: 0.0
   - disable: 'N'
   - hosp_visit: 'N'
   - er_visit: 'N'

2. **Merge the Symptoms into Singular Column**
   - Create a new column that holds a flattened list of all symptoms

3. **Create a new boolean column for every top allergy, other diagnosis, and medication**

4. **Create 'Covid' Target Column**
   - There are two symptoms that encompass whether or not someone got covid 'SARS-CoV-2 test positive' and'COVID-19'.
   - Need to merge those two columns together and return True if either of them are True

5. **Drop 'SARS-CoV-2 test positive', 'COVID-19' and 'All Symptoms' Columns**

3. Process Data

```
pipe_symp = Pipeline([('fillNaNs',          FillNaNs(list(fillNa.keys()), list(fillNa.values()))),          # Fill specific NA columns
                      ('Merge Symptoms',    MergeSymptomDFs(symptoms, 'id', final_cols[:2], final_cols[1:])), # Merge the symptoms into singular column
                      ('Expand Symptoms',   ConvertColumnLists(covid_cols[:2], final_cols[0])),             # Create a new boolean column for covid-19/cov test pos
                      ('Expand History',    ConvertColumnLists(history_list, 'history')),                   # Create a new boolean column for every top symptom
                      ('Expand Meds',       ConvertColumnLists(meds, 'other_meds')),                        # Create a new boolean column for every top medication
                      ('Expand Allerigies', ConvertColumnLists(meds, 'allergies')),                         # Create a new boolean column for every top allergy
                      ('Create Target',     BooleanInAnyDF(covid_cols[:2], True, covid_cols[-1])),          # Create 'covid' target column
                      ('Drop Columns',      DropColumns(covid_cols[:3] + text)),                            # Drop unnecessary columns
                     ])

df = pipe_symp.fit_transform(df)
```

	id	manu	dose	route	state	age	sex	hospdays	disable	recvd	...	other_meds	history	hosp_visit	er_visit	allergies	symptom1	symptom2	symptom3	symptom4	symptom5
0	916600	MODERNA	1	IM	TX	33.0	F	0.0	N	Y	...	None	None	Y	N	Pcn and bee venom	Dysphagia	Epiglottitis	NaN	NaN	NaN
1	916601	MODERNA	1	IM	CA	73.0	F	0.0	N	Y	...	Patient residing at nursing facility. See pati...	Patient residing at nursing facility. See pati...	Y	N	"Dairy"	Anxiety	Dyspnoea	NaN	NaN	NaN
2	916602	PFIZER\BIONTECH	1	IM	WA	23.0	F	0.0	N	U	...	None	None	N	Y	Shellfish	Chest discomfort	Dysphagia	Pain in extremity	Visual impairment	NaN
3	916604	MODERNA	1	IM	TX	47.0	F	0.0	N	N	...	Na	None	N	N	Na	Injection site erythema	Injection site pruritus	Injection site swelling	Injection site warmth	NaN
4	916606	MODERNA	1	IM	NV	44.0	F	0.0	N	Y	...	None	None	N	N	iodine (shellfish) has epipen	Pharyngeal swelling	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
18107	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	N	...	Meds prior to admission: Amiodarone, ASA, Lipi...	Arthritis, Afib, chronic anticoagulation, chro...	N	N	Pneumococcal vaccine - rash	Malaise	Nausea	Oxygen therapy	Parosmia	Productive cough
18108	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	N	...	Meds prior to admission: Amiodarone, ASA, Lipi...	Arthritis, Afib, chronic anticoagulation, chro...	N	N	Pneumococcal vaccine - rash	SARS-CoV-2 test positive	Streptococcus test negative	Taste disorder	Vomiting	NaN
18109	1057281	MODERNA	1	IM	MD	77.0	F	0.0	N	N	...	None	None	N	N	None	Death	NaN	NaN	NaN	NaN
18110	1057348	PFIZER\BIONTECH	1	IM	TX	88.0	F	0.0	N	N	...	lasik, blood pressure	congestive heart failure	Y	N	none	Death	Dysarthria	Dysstasia	Fatigue	Feeding disorder
18111	1057795	MODERNA	1	IM	OH	82.0	F	0.0	N	N	...	None	None	N	N	None	Death	NaN	NaN	NaN	NaN

18112 rows × 23 columns

3. Process Data

```
pipe_symp = Pipeline([('fillNaNs',          FillNaNs(list(fillNa.keys()), list(fillNa.values())),          # Fill specific NA columns
                      ('Merge Symptoms',    MergeSymptomDFs(symptoms, 'id', final_cols[:2], final_cols[1:])), # Merge the symptoms into singular column
                      ('Expand Symptoms',   ConvertColumnLists(covid_cols[:2], final_cols[0])),            # Create a new boolean column for covid-19/cov test pos
                      ('Expand History',    ConvertColumnLists(history_list, 'history')),                  # Create a new boolean column for every top symptom
                      ('Expand Meds',       ConvertColumnLists(meds, 'other_meds')),                       # Create a new boolean column for every top medication
                      ('Expand Allerigies', ConvertColumnLists(meds, 'allergies')),                        # Create a new boolean column for every top allergy
                      ('Create Target',     BooleanInAnyDF(covid_cols[:2], True, covid_cols[-1])),         # Create 'covid' target column
                      ('Drop Columns',      DropColumns(covid_cols[:3] + text)),                           # Drop unnecessary columns
                     ])

df = pipe_symp.fit_transform(df)
```

	all_symptoms	id	manu	dose	route	state	age	sex	hospdays	disable	history	numdays	adminby	hosp_visit	er_visit	other_meds	allergies
0	NaN	916600	MODERNA	1	IM	TX	33.0	F	0.0	N	None	2.0	PVT	Y	N	None	Pcn and bee venom
1	NaN	916601	MODERNA	1	IM	CA	73.0	F	0.0	N	Patient residing at nursing facility. See pati...	0.0	SEN	Y	N	Patient residing at nursing facility. See pati...	"Dairy"
2	Chest discomfort, Dysphagia, Pain in extremity...	916602	PFIZER\BIONTECH	1	IM	WA	23.0	F	0.0	N	None	0.0	SEN	N	Y	None	Shellfish
3	Injection site erythema, Injection site prurit...	916604	MODERNA	1	IM	TX	47.0	F	0.0	N	None	7.0	PUB	N	N	Na	Na
4	NaN	916606	MODERNA	1	IM	NV	44.0	F	0.0	N	None	0.0	PVT	N	N	None	iodine (shellfish) has epipen
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
38157	SARS-CoV-2 test positive, Streptococcus test n...	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	Arthritis, Afib, chronic anticoagulation, chro...	9.0	PVT	N	N	Meds prior to admission: Amiodarone, ASA, Lipi...	Pneumococcal vaccine - rash
38158	SARS-CoV-2 test positive, Streptococcus test n...	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	Arthritis, Afib, chronic anticoagulation, chro...	9.0	PVT	N	N	Meds prior to admission: Amiodarone, ASA, Lipi...	Pneumococcal vaccine - rash
38159	NaN	1057281	MODERNA	1	IM	MD	77.0	F	0.0	N	None	14.0	OTH	N	N	None	None
38160	Death, Dysarthria, Dysstasia, Fatigue	1057348	PFIZER\BIONTECH	1	IM	TX	88.0	F	0.0	N	congestive heart failure	2.0	PVT	Y	N	lasik, blood pressure	none
38161	NaN	1057795	MODERNA	1	IM	OH	82.0	F	0.0	N	None	33.0	PUB	N	N	None	None

38162 rows × 17 columns

3. Process Data

```
pipe_symp = Pipeline([('fillNaNs',          FillNaNs(list(fillNa.keys()), list(fillNa.values())),          # Fill specific NA columns
                      ('Merge Symptoms',    MergeSymptomDFs(symptoms, 'id', final_cols[:2], final_cols[1:])), # Merge the symptoms into singular column
                      ('Expand Symptoms',   ConvertColumnLists(covid_cols[:2], final_cols[0])),            # Create a new boolean column for covid-19/cov test pos
                      ('Expand History',    ConvertColumnLists(history_list, 'history')),                  # Create a new boolean column for every top symptom
                      ('Expand Meds',       ConvertColumnLists(meds, 'other_meds')),                       # Create a new boolean column for every top medication
                      ('Expand Allerigies', ConvertColumnLists(meds, 'allergies')),                        # Create a new boolean column for every top allergy
                      ('Create Target',     BooleanInAnyDF(covid_cols[:2], True, covid_cols[-1])),         # Create 'covid' target column
                      ('Drop Columns',      DropColumns(covid_cols[:3] + text)),                           # Drop unnecessary columns
                     ])

df = pipe_symp.fit_transform(df)
```

	all_symptoms	id	manu	dose	route	state	age	sex	hospdays	disable	history	numdays	adminby	hosp_visit	er_visit	other_meds	allergies
0	NaN	916600	MODERNA	1	IM	TX	33.0	F	0.0	N	None	2.0	PVT	Y	N	None	Pcn and bee venom
1	NaN	916601	MODERNA	1	IM	CA	73.0	F	0.0	N	Patient residing at nursing facility. See pati...	0.0	SEN	Y	N	Patient residing at nursing facility. See pati...	"Dairy"
2	Chest discomfort, Dysphagia, Pain in extremity...	916602	PFIZER\BIONTECH	1	IM	WA	23.0	F	0.0	N	None	0.0	SEN	N	Y	None	Shellfish
3	Injection site erythema, Injection site prurit...	916604	MODERNA	1	IM	TX	47.0	F	0.0	N	None	7.0	PUB	N	N	Na	Na
4	NaN	916606	MODERNA	1	IM	NV	44.0	F	0.0	N	None	0.0	PVT	N	N	None	iodine (shellfish) has epipen
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
38157	SARS-CoV-2 test positive, Streptococcus test n...	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	Arthritis, Afib, chronic anticoagulation, chro...	9.0	PVT	N	N	Meds prior to admission: Amiodarone, ASA, Lipi...	Pneumococcal vaccine - rash
38158	SARS-CoV-2 test positive, Streptococcus test n...	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	Arthritis, Afib, chronic anticoagulation, chro...	9.0	PVT	N	N	Meds prior to admission: Amiodarone, ASA, Lipi...	Pneumococcal vaccine - rash
38159	NaN	1057281	MODERNA	1	IM	MD	77.0	F	0.0	N	None	14.0	OTH	N	N	None	None
38160	Death, Dysarthria, Dysstasia, Fatigue	1057348	PFIZER\BIONTECH	1	IM	TX	88.0	F	0.0	N	congestive heart failure	2.0	PVT	Y	N	lasik, blood pressure	none
38161	NaN	1057795	MODERNA	1	IM	OH	82.0	F	0.0	N	None	33.0	PUB	N	N	None	None

38162 rows × 17 columns

3. Process Data

```
pipe_symp = Pipeline([('fillNaNs',           FillNaNs(list(fillNa.keys()), list(fillNa.values())),          # Fill specific NA columns
                      ('Merge Symptoms',     MergeSymptomDFs(symptoms, 'id', final_cols[:2], final_cols[1:])), # Merge the symptoms into singular column
                      ('Expand Symptoms',    ConvertColumnLists(covid_cols[:2], final_cols[0])),            # Create a new boolean column for covid-19/cov test pos
                      ('Expand History',     ConvertColumnLists(history_list, 'history')),                  # Create a new boolean column for every top symptom
                      ('Expand Meds',        ConvertColumnLists(meds, 'other_meds')),                       # Create a new boolean column for every top medication
                      ('Expand Allerigies',  ConvertColumnLists(meds, 'allergies')),                        # Create a new boolean column for every top allergy
                      ('Create Target',      BooleanInAnyDF(covid_cols[:2], True, covid_cols[-1])),         # Create 'covid' target column
                      ('Drop Columns',       DropColumns(covid_cols[:3] + text)),                           # Drop unnecessary columns
                     ])

df = pipe_symp.fit_transform(df)
```

	all_symptoms	id	manu	dose	route	state	age	sex	hospdays	disable	...	gerd	high blood pressure	depression	levothyroxine	multivitamin	tylenol	adderall	ibuprofen	prenatal	naltrexone
0	NaN	916600	MODERNA	1	IM	TX	33.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
1	NaN	916601	MODERNA	1	IM	CA	73.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
2	Chest discomfort, Dysphagia, Pain in extremity...	916602	PFIZER\BIONTECH	1	IM	WA	23.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
3	Injection site erythema, Injection site prurit...	916604	MODERNA	1	IM	TX	47.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
4	NaN	916606	MODERNA	1	IM	NV	44.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
38157	SARS-CoV-2 test positive, Streptococcus test n...	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	...	True	False	False	False	False	False	False	False	False	False
38158	SARS-CoV-2 test positive, Streptococcus test n...	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	...	True	False	False	False	False	False	False	False	False	False
38159	NaN	1057281	MODERNA	1	IM	MD	77.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
38160	Death, Dysarthria, Dysstasia, Fatigue	1057348	PFIZER\BIONTECH	1	IM	TX	88.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
38161	NaN	1057795	MODERNA	1	IM	OH	82.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False

38162 rows × 34 columns

3. Process Data

```
pipe_symp = Pipeline([('fillNaNs',          FillNaNs(list(fillNa.keys()), list(fillNa.values()))),          # Fill specific NA columns
                      ('Merge Symptoms',    MergeSymptomDFs(symptoms, 'id', final_cols[:2], final_cols[1:])), # Merge the symptoms into singular column
                      ('Expand Symptoms',   ConvertColumnLists(covid_cols[:2], final_cols[0])),             # Create a new boolean column for covid-19/cov test pos
                      ('Expand History',    ConvertColumnLists(history_list, 'history')),                   # Create a new boolean column for every top symptom
                      ('Expand Meds',       ConvertColumnLists(meds, 'other_meds')),                        # Create a new boolean column for every top medication
                      ('Expand Allerigies', ConvertColumnLists(meds, 'allergies')),                         # Create a new boolean column for every top allergy
                      ('Create Target',     BooleanInAnyDF(covid_cols[:2], True, covid_cols[-1])),          # Create 'covid' target column
                      ('Drop Columns',      DropColumns(covid_cols[:3] + text)),                            # Drop unnecessary columns
                     ])

df = pipe_symp.fit_transform(df)
```

	all_symptoms	id	manu	dose	route	state	age	sex	hospdays	disable	...	high blood pressure	depression	levothyroxine	multivitamin	tylenol	adderall	ibuprofen	prenatal	naltrexone	COVID
0	NaN	916600	MODERNA	1	IM	TX	33.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
1	NaN	916601	MODERNA	1	IM	CA	73.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
2	Chest discomfort, Dysphagia, Pain in extremity...	916602	PFIZER\BIONTECH	1	IM	WA	23.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
3	Injection site erythema, Injection site pruri...	916604	MODERNA	1	IM	TX	47.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
4	NaN	916606	MODERNA	1	IM	NV	44.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
38157	SARS-CoV-2 test positive, Streptococcus test n...	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	...	False	False	False	False	False	False	False	False	False	True
38158	SARS-CoV-2 test positive, Streptococcus test n...	1057082	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	...	False	False	False	False	False	False	False	False	False	True
38159	NaN	1057281	MODERNA	1	IM	MD	77.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
38160	Death, Dysarthria, Dysstasia, Fatigue	1057348	PFIZER\BIONTECH	1	IM	TX	88.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False
38161	NaN	1057795	MODERNA	1	IM	OH	82.0	F	0.0	N	...	False	False	False	False	False	False	False	False	False	False

38162 rows × 35 columns

3. Process Data

```
pipe_symp = Pipeline([('fillNaNs',          FillNaNs(list(fillNa.keys()), list(fillNa.values()))),          # Fill specific NA columns
                      ('Merge Symptoms',    MergeSymptomDFs(symptoms, 'id', final_cols[:2], final_cols[1:])), # Merge the symptoms into singular column
                      ('Expand Symptoms',   ConvertColumnLists(covid_cols[:2], final_cols[0])),             # Create a new boolean column for covid-19/cov test pos
                      ('Expand History',    ConvertColumnLists(history_list, 'history')),                   # Create a new boolean column for every top symptom
                      ('Expand Meds',       ConvertColumnLists(meds, 'other_meds')),                        # Create a new boolean column for every top medication
                      ('Expand Allerigies', ConvertColumnLists(meds, 'allergies')),                         # Create a new boolean column for every top allergy
                      ('Create Target',     BooleanInAnyDF(covid_cols[:2], True, covid_cols[-1])),          # Create 'covid' target column
                      ('Drop Columns',      DropColumns(covid_cols[:3] + text)),                            # Drop unnecessary columns
                     ])

df = pipe_symp.fit_transform(df)
```

	manu	dose	route	state	age	sex	hospdays	disable	numdays	adminby	...	high blood pressure	depression	levothyroxine	multivitamin	tylenol	adderall	ibuprofen	prenatal	naltrexone	COVID
0	MODERNA	1	IM	TX	33.0	F	0.0	N	2.0	PVT	...	False	False	False	False	False	False	False	False	False	False
1	MODERNA	1	IM	CA	73.0	F	0.0	N	0.0	SEN	...	False	False	False	False	False	False	False	False	False	False
2	PFIZER\BIONTECH	1	IM	WA	23.0	F	0.0	N	0.0	SEN	...	False	False	False	False	False	False	False	False	False	False
3	MODERNA	1	IM	TX	47.0	F	0.0	N	7.0	PUB	...	False	False	False	False	False	False	False	False	False	False
4	MODERNA	1	IM	NV	44.0	F	0.0	N	0.0	PVT	...	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
38157	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	9.0	PVT	...	False	False	False	False	False	False	False	False	False	True
38158	PFIZER\BIONTECH	1	IM	KY	86.0	M	12.0	N	9.0	PVT	...	False	False	False	False	False	False	False	False	False	True
38159	MODERNA	1	IM	MD	77.0	F	0.0	N	14.0	OTH	...	False	False	False	False	False	False	False	False	False	False
38160	PFIZER\BIONTECH	1	IM	TX	88.0	F	0.0	N	2.0	PVT	...	False	False	False	False	False	False	False	False	False	False
38161	MODERNA	1	IM	OH	82.0	F	0.0	N	33.0	PUB	...	False	False	False	False	False	False	False	False	False	False

38162 rows × 28 columns

3. Process Data

```
con_pipe = Pipeline([('imputer', SimpleImputer(strategy='median', add_indicator=True)), # replaces the NaN values with median
                     ('scaler', StandardScaler())]) # standardize the numerical features

cat_pipe = Pipeline([('imputer', SimpleImputer(strategy='most_frequent')), # replaces the NaN values with most frequent
                     ('encoder', OneHotEncoder(handle_unknown='ignore'))]) # represent categorical variables as binary vectors

final_prep_pipe = ColumnTransformer([('categorical', cat_pipe,   (X.dtypes == object)),
                                     ('continuous',  con_pipe, ~(X.dtypes == object))])
```

```
X = final_prep_pipe.fit_transform(X)
y = y.values
```

3. Process Data

Ensure the Same Ratio of Classes for Train, Val and Test

```
stratSplit = StratifiedShuffleSplit(n_splits=1, test_size=0.2, random_state=42)

for train_idx, test_idx in stratSplit.split(X, y):
    X_train=X[train_idx]
    y_train=y[train_idx]
    X_test=X[test_idx]
    y_test=y[test_idx]

for train_idx, val_idx in stratSplit.split(X_train, y_train):
    X_val= X_train[val_idx]
    y_val= y_train[val_idx]
    X_train=X_train[train_idx]
    y_train=y_train[train_idx]
```

4. Model Selection: Metric

Balanced Accuracy:
Since we have imbalanced data, balanced accuracy score is best to account for both covid positive and not covid positive outcome classes.

4. Model Selection: Unbalanced Data

	Regular	SMOTE	Random Oversampling
KNeighbors :	0.548	0.638	0.684
RF Classifier :	0.576	0.708	0.866
SGD Classifier :	0.5	0.585	0.585
Extra Trees Classifier:	0.557	0.694	0.861
AdaBoost Classifier :	0.5	0.606	0.669
MLP Classifier :	0.561	0.734	0.839
Logistic Classifier :	0.5	0.619	0.598

4. Model Selection

Randomized Search CV / Hyperparameter Tuning

Unlike GridSearchCV, RandomizedSearchCV

randomly searches on hyperparameters

whereas GridSearchCV tries all parameters

```
estimators = {
    RandomForestClassifier: {
        'est__n_estimators': [100,200,300,500],
        'est__min_samples_leaf': [3,4,5]
    },
    ExtraTreesClassifier: {
        'est__n_estimators': [100,200,300,500],
        'est__min_samples_leaf': [3,4,5]
    },
    MLPClassifier: {
        'est__tol': [1e-3, 1e-4, 1e-5],
        'est__alpha': [1e-3, 1e-4, 1e-5],
        'est__solver': ['adam', 'sgd'],
        'est__max_iter':[1000]
    }
}

scores = []
for estimator, param_dist in estimators.items():

    pipe = Pipeline([('est', estimator())])

    rscv = RandomizedSearchCV(
        pipe,
        param_dist,
        n_jobs=-1,
    )
    rscv.fit(X, y)
    clean_best_param = {
        k.replace('est__', ''): v
        for k, v in rscv.best_params_.items()
    }

    pipe = Pipeline([('est', estimator(**clean_best_param))])

    pipe.fit(X_oversample, y_oversample)
    y_pred = pipe.predict(X_val)
    score = balanced_accuracy_score(y_val, y_pred)
    scores.append((estimator.__name__, clean_best_param, score))
```

4. Model Selection

Randomized Search CV / Hyperparameter Tuning

Unlike GridSearchCV, RandomizedSearchCV

randomly searches on hyperparameters

whereas GridSearchCV tries all parameters

```
[('RandomForestClassifier',
  {'n_estimators': 100, 'min_samples_leaf': 5},
  0.8470458470825627),
 ('ExtraTreesClassifier',
  {'n_estimators': 100, 'min_samples_leaf': 5},
  0.7933339758564297),
 ('MLPClassifier',
  {'tol': 0.001, 'max_iter': 1000, 'alpha': 0.01},
  0.8237826864502588)]
```

```
estimators = {
    RandomForestClassifier: {
        'est__n_estimators': [100,200,300,500],
        'est__min_samples_leaf': [3,4,5]
    },
    ExtraTreesClassifier: {
        'est__n_estimators': [100,200,300,500],
        'est__min_samples_leaf': [3,4,5]
    },
    MLPClassifier: {
        'est__tol': [1e-3, 1e-4, 1e-5],
        'est__alpha': [1e-3, 1e-4, 1e-5],
        'est__solver': ['adam', 'sgd'],
        'est__max_iter':[1000]
    }
}

scores = []
for estimator, param_dist in estimators.items():

    pipe = Pipeline([('est', estimator())])

    rscv = RandomizedSearchCV(
        pipe,
        param_dist,
        n_jobs=-1,
    )
    rscv.fit(X, y)
    clean_best_param = {
        k.replace('est__', ''): v
        for k, v in rscv.best_params_.items()
    }

    pipe = Pipeline([('est', estimator(**clean_best_param))])

    pipe.fit(X_oversample, y_oversample)
    y_pred = pipe.predict(X_val)
    score = balanced_accuracy_score(y_val, y_pred)
    scores.append((estimator.__name__, clean_best_param, score))
```

5. Deliver

Validation set

Analysis: Random Forest Performed Best

```
pipe = Pipeline([('est', RandomForestClassifier(n_estimators=100,
                                                 min_samples_leaf = 2,
                                                 random_state=42))])

pipe.fit(X_oversample, y_oversample)
y_pred = pipe.predict(X_val)
print(f"RF Best Model: {round(balanced_accuracy_score(y_val, y_pred),3)}")
```

```
RF Best Model: 0.866
```

5. Deliver

Test set: truest measure of generality

```
pipe = Pipeline([('est', RandomForestClassifier(n_estimators=100,
                                                min_samples_leaf = 2,
                                                random_state=42))])

pipe.fit(X_oversample, y_oversample)
y_pred = pipe.predict(X_test)
print(f"RF Best Model: {round(balanced_accuracy_score(y_test, y_pred),3)}\n")
```

```
RF Best Model: 0.818
```

Summary

1. With ~ 81.8% balanced accuracy, we can predict whether or not someone who receives a vaccine will still get covid
2. The best model turned out to be random forest, possibly due to the fact that it is robust to outliers and feature's that aren't as important will simply not get chosen during feature selection
3. Despite class imbalance, we were able to get the model to perform fairly well due to oversampling techniques.
4. This is a useful model for those who may feel uncertain about getting the vaccine and wants to know their chances of getting covid even after receiving the vaccine

Limitations

1. Model only used the top allergies, diseases, and medication. The data pipeline dropped all others that may have had higher predictive power.
2. A lot of other variables that would be predictive of whether or not someone still gets covid is missing (e.g, whether or not someone lives in a city).
3. **Next steps:** Continue working on feature engineering. Carry out methods to measure feature importance and encode more allergies/diseases/medication.