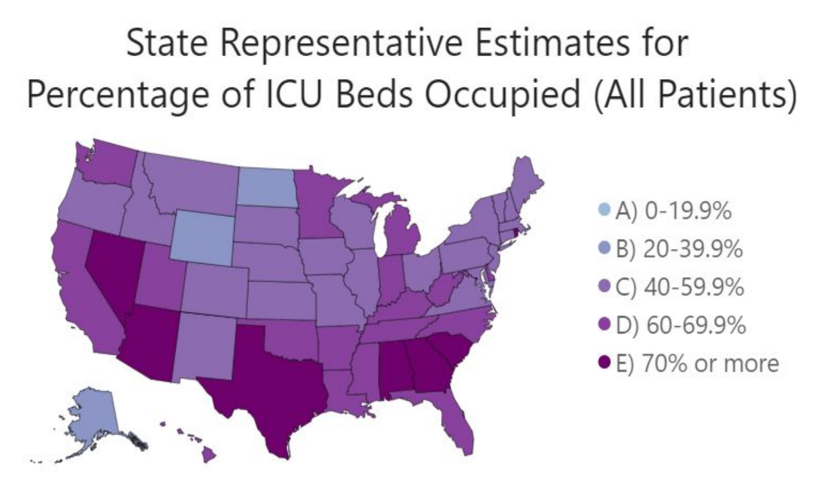
PredictCOVID

By Gianna, Vienna, Jessica, Christina, Srihita, Fiona

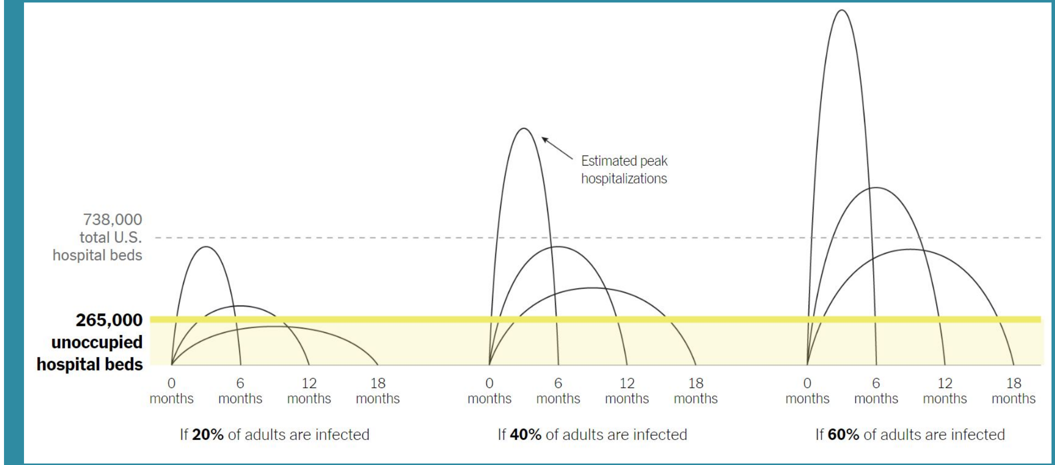
Co-TAs: Alice, Bianca

Infected Patients Overwhelm COVID-19 Testing and ICU Beds



State Representative Estimates for Percentage of ICU Beds Occupied (All Patients)

Source: CDC.gov

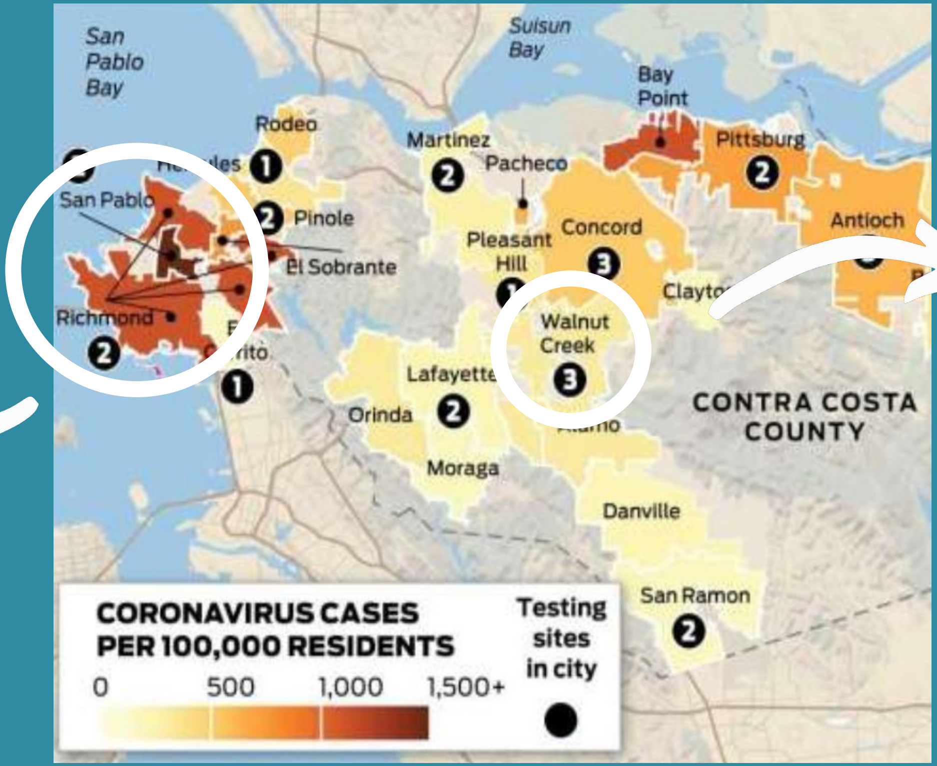


Source: NY Times. Harvard Global Health Institute.

The supply of COVID-19 tests play a big role in controlling the pandemic.

Social Inequalities in Healthcare

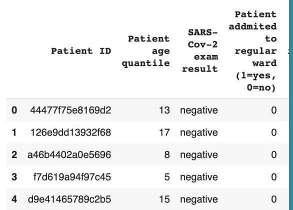
Richmond and San Pablo (predominantly Latino, Asian American and Black working-class cities) have some of the highest infection rates in Contra Costa County



But have same number of community testing sites as Walnut Creek (an affluent, mostly white city with half the population)

Source: SFChronicle.com

Drumroll please….

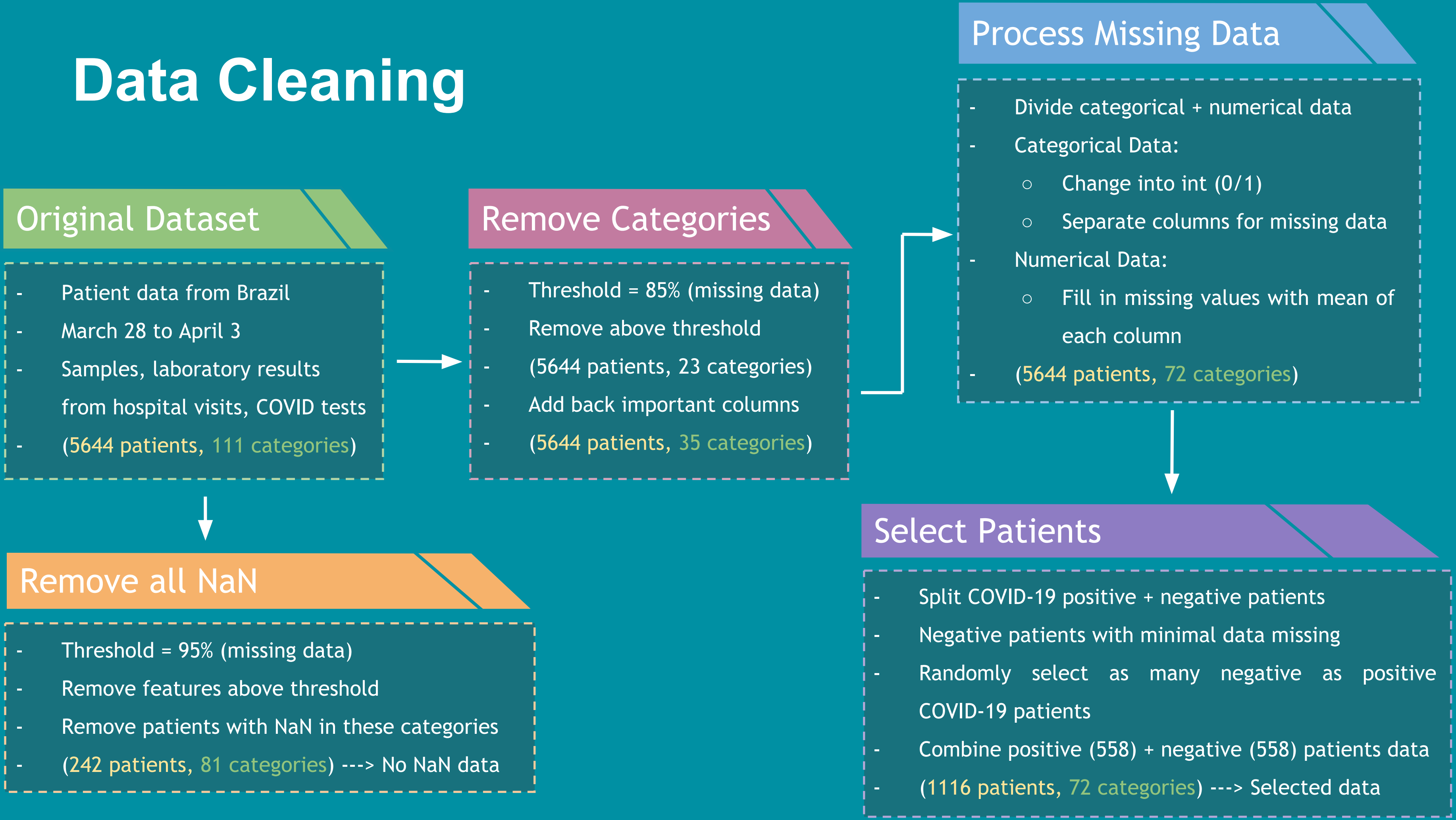


	Patient ID	Patient age quantile	SARS-Cov-2 exam result	Patient addmited to regular ward (1=yes, 0=no)
0	44477f75e8169d2	13	negative	0
1	126e9dd13932f68	17	negative	0
2	a46b4402a0e5696	8	negative	0
3	f7d619a94f97c45	5	negative	0
4	d9e41465789c2b5	15	negative	0

PredictCOVID

COVID Status & COVID Severity!

Data Cleaning



Original Dataset

- Patient data from Brazil
- March 28 to April 3
- Samples, laboratory results from hospital visits, COVID tests
- (5644 patients, 111 categories)

Remove Categories

- Threshold = 85% (missing data)
- Remove above threshold
- (5644 patients, 23 categories)
- Add back important columns
- (5644 patients, 35 categories)

Process Missing Data

- Divide categorical + numerical data
- Categorical Data:
  - Change into int (0/1)
  - Separate columns for missing data
- Numerical Data:
  - Fill in missing values with mean of each column
- (5644 patients, 72 categories)

Select Patients

- Split COVID-19 positive + negative patients
- Negative patients with minimal data missing
- Randomly select as many negative as positive COVID-19 patients
- Combine positive (558) + negative (558) patients data
- (1116 patients, 72 categories) ---> Selected data

Remove all NaN

- Threshold = 95% (missing data)
- Remove features above threshold
- Remove patients with NaN in these categories
- (242 patients, 81 categories) ---> No NaN data

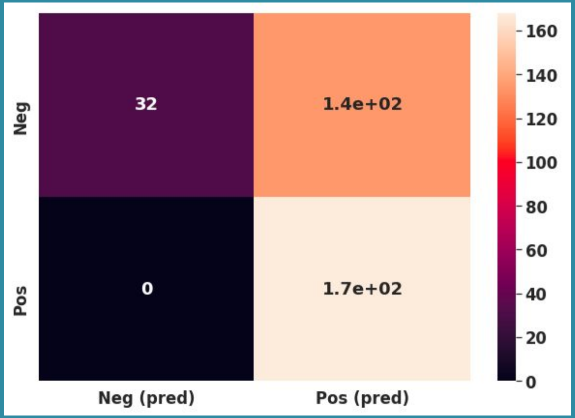
Status ML Model

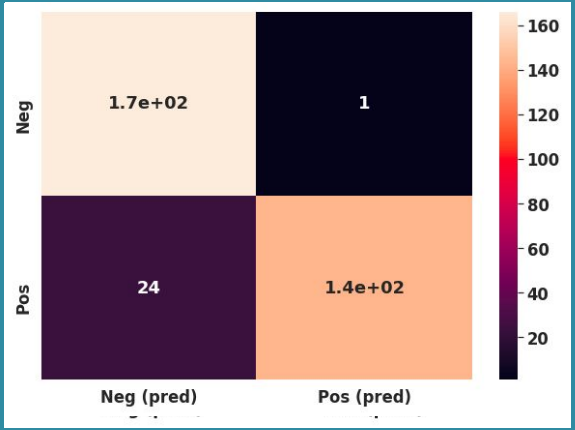
NoNAN vs Selected Data Accuracy of Different Classifiers

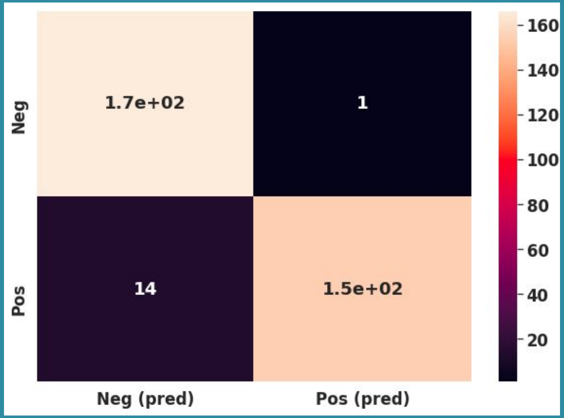
Classifier	K Nearest Neighbors	Random Forest	Naive Bayes	Gaussian Process	SVM	Decision Tree	QDA	MLP
Accuracy Score (NoNAN data)	89.0%	87.7%	42.5%	92.1%	82.2%	83.6%	78.1%	89.0%
Accuracy Score (Selected data)	88.7%	90.7% (93.0% with feature importance)	92.5%	91.3%	92.8%	93.4%	59.7%	95.5%

→ The accuracy scores differed with respect to dataset and classifier. Most accuracies increased.

QDA, NB, MLP, Oh My!







QDA

59.7%

NB

92.5%

MLP

95.5%

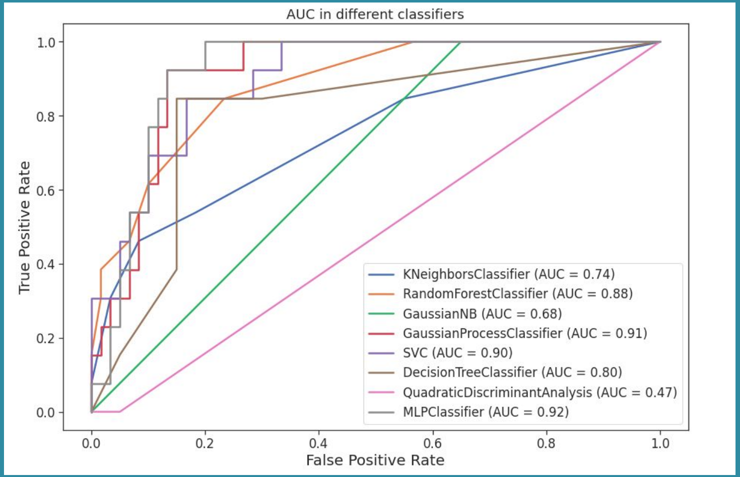
To have COVID or not to have COVID: Consequences of false negatives vs. false positives

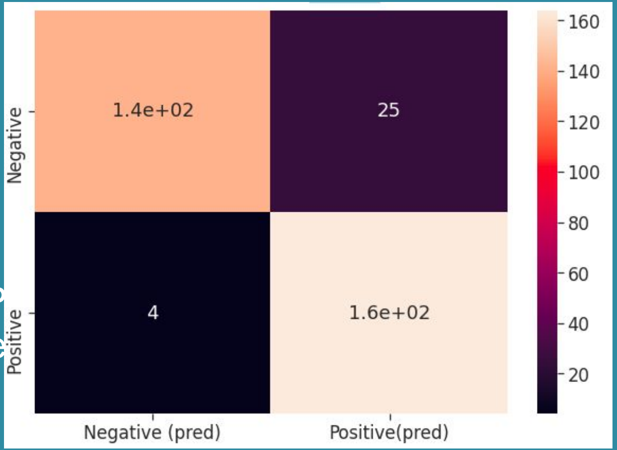
False Negatives	False Positives
- Not receive treatments in time - Keep spreading COVID-19 - Unknowingly infect others around - More severe	- Take up limited resources - Tests, hospitals, ICU - Small scale → more preferable

Our Status Model Choice

91.3%
Accuracy

Gaussian Process





There are a lot less false negatives from using the GP model. Although the accuracy isn't the highest and we do run the risk of unnecessarily using up resources, we prevent the spread of COVID more.

Severity ML Models

Chart of Classifier Values

Classifier	K Nearest Neighbors	Random Forest	Naive Bayes	Gaussian Process	SVM	Decision Tree	QDA	MLP
Accuracy Score (NoNAN data)	72.6%	74.0%	20.5%	69.0%	69.9%	69.9%	68.5%	68.5%

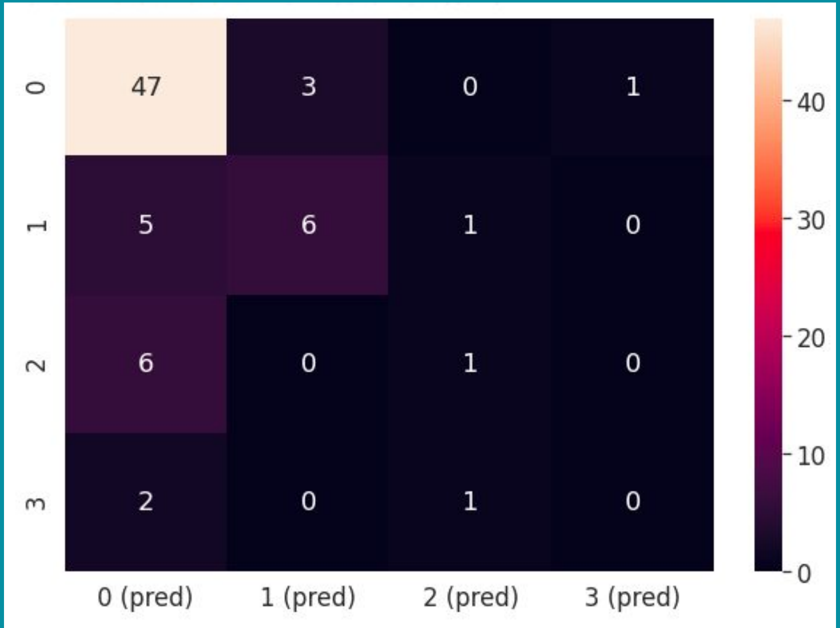
0, 1, 2, 3, which level of treatment is right for me?

0: no treatment

1: regular treatment

2: semi-intensive treatment

3: intensive care unit (ICU)



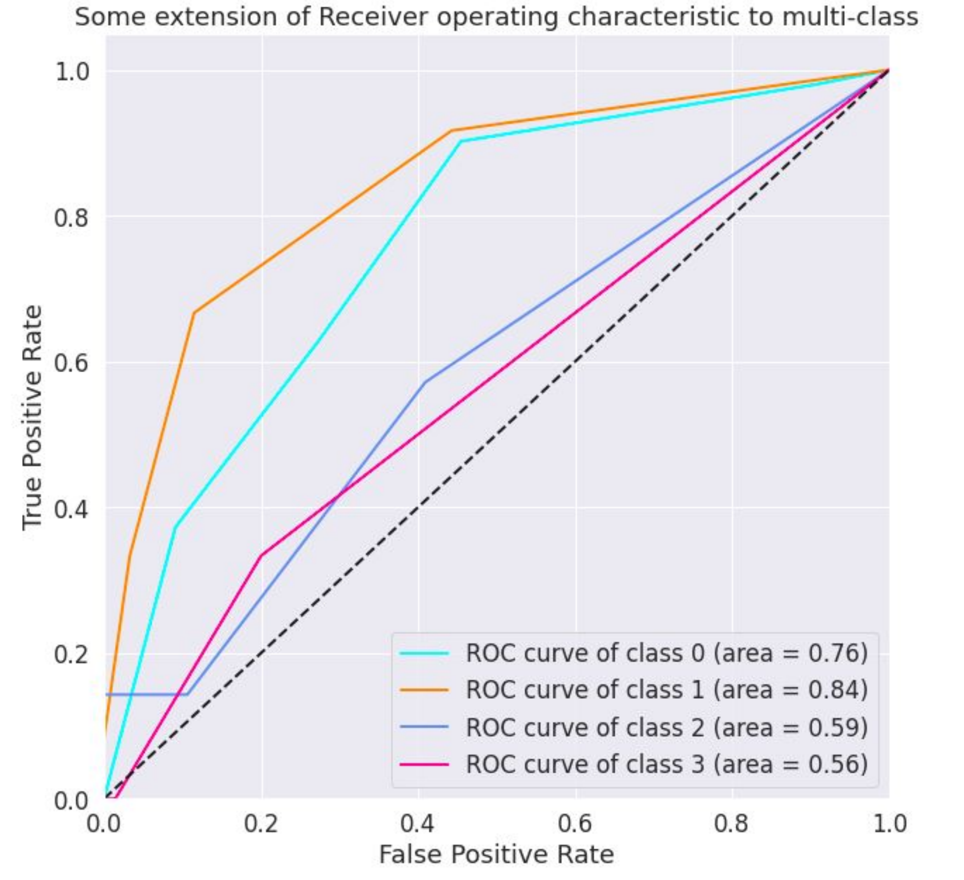
	0 (pred)	1 (pred)	2 (pred)	3 (pred)
0	47	3	0	1
1	5	6	1	0
2	6	0	1	0
3	2	0	1	0

Random Forest Classifier

Falsely Predicted: 3 Actual: 0	Falsely Predicted: 0 Actual: 3
- waste resources - takes up limited space (ICU)	- Not receive treatments in time - Conditions worsen

ROC/AUC Graphs for Severity

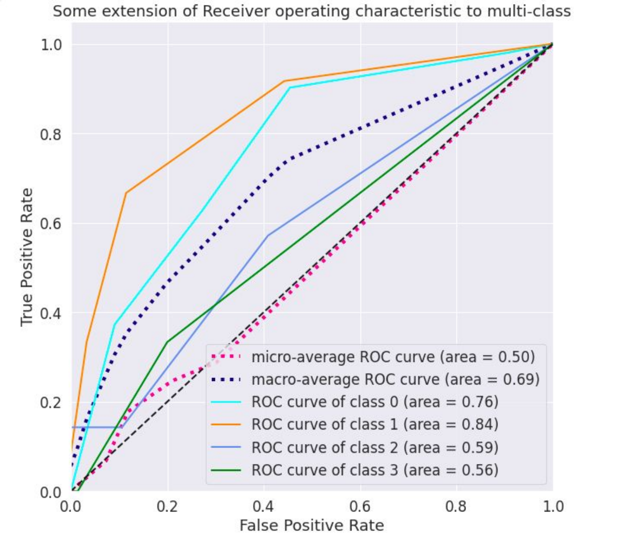
0: no treatment
1: regular treatment
2: semi-intensive treatment
3: intensive care unit (ICU)

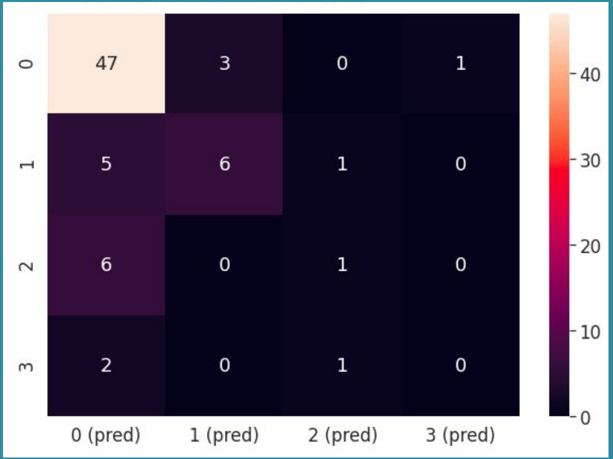


Our Severity Model Choice

74%
Accuracy

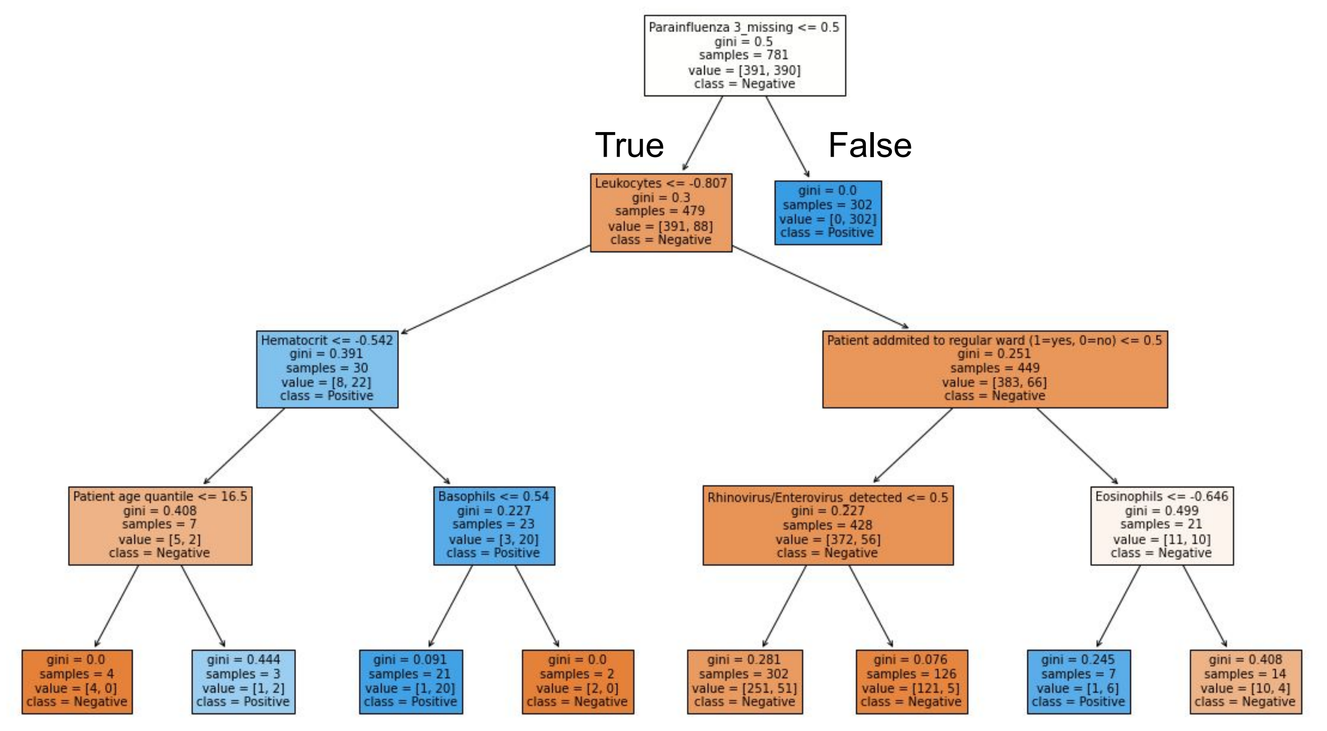
Random Forest





	0 (pred)	1 (pred)	2 (pred)	3 (pred)
0	47	3	0	1
1	5	6	1	0
2	6	0	1	0
3	2	0	1	0

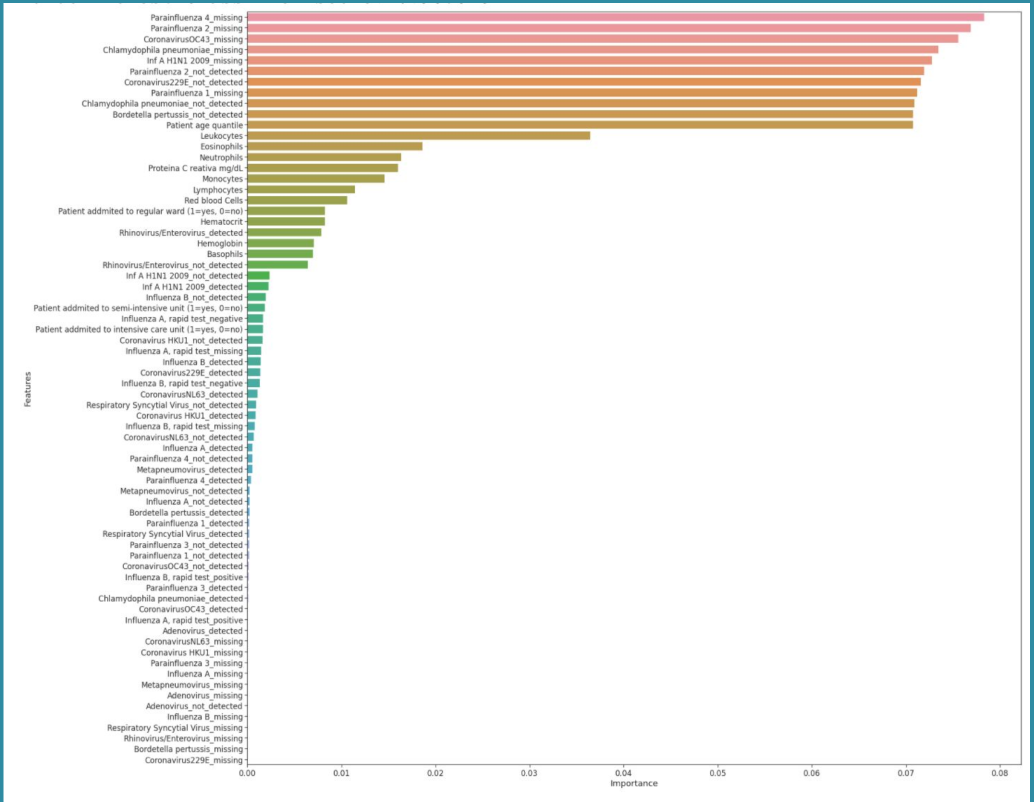
Decision Tree for Selected Data shows that many immune related features are important



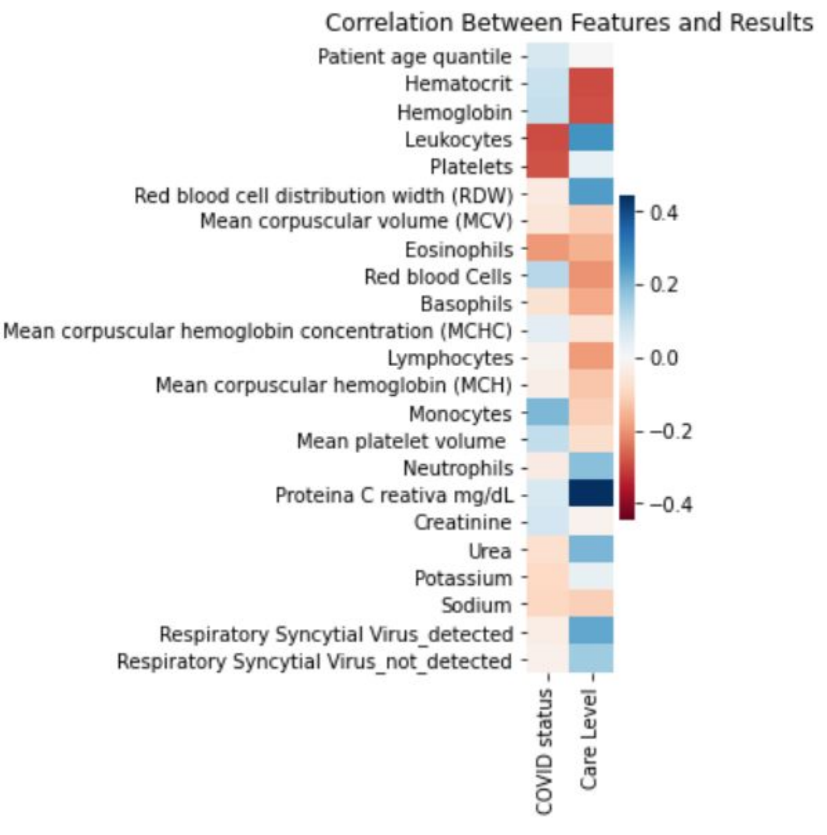
- Leukocytes
- Hematocrit
- Age quantile
- Basophils
- Eosinophils
- Rhinovirus/enterovirus
- Patient admitted to regular ward

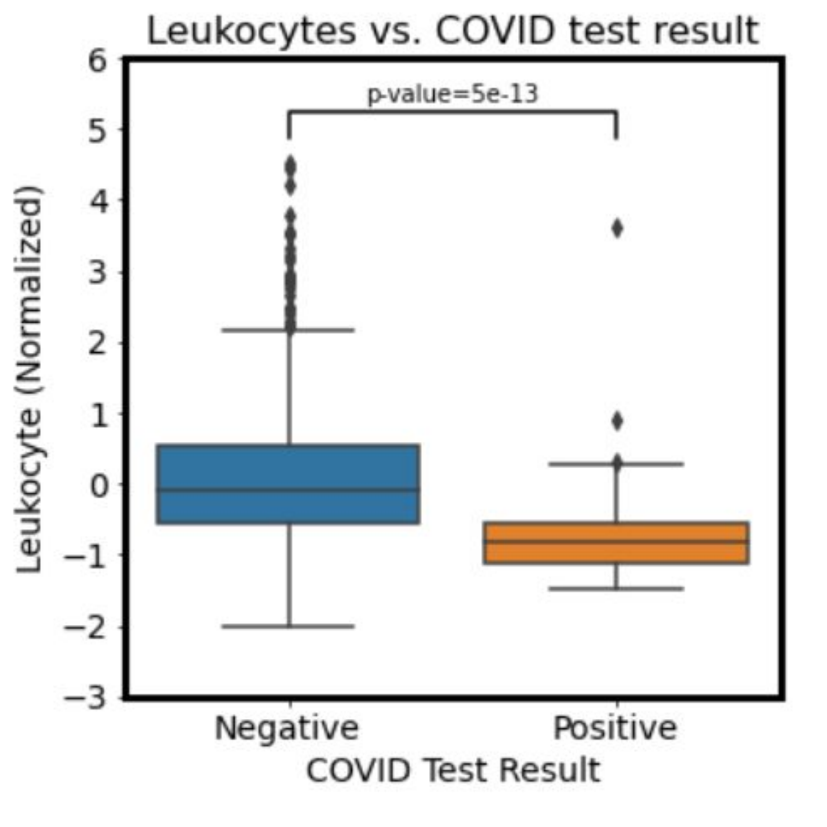
Feature Importance Using Random Forest Classifier

1. Parainfluenza 2_not_detected
2. Coronavirus229E_not_detected
3. Chlamydophila pneumonia_not_detected
4. Bordetella pertussis_not_detected
5. Patient Age Quantile
6. Leukocytes
7. Eosinophils
8. Neutrophils
9. C Reactive Protein
10. Monocytes



Usage in Clinical Decision Making





Bias and Limitations

Data	Location
- Missing data - Personal / family medical history - Lifestyle habits - Prior exposures - Specific age - Uneven ratio between COVID negative + positive patient data	- Hospital Israelita Albert Einstein - São Paulo, Brazil - Date (7 days) - Quarantine - Not full representation of Brazil or the world

Conclusions and Future Directions

- Best Model for Status
  - Gaussian Process Classifier (91.3%)
- Best Model for Severity
  - Random Forest (74% accuracy)
- Important features were highly immune related.
- Future Plans
  - Validate and refine model
  - Include more features such as pre-existing conditions/current medications
  - Run models using data from different hospitals
  - Main goal: implement this model in a hospital

Acknowledgments

AI4ALL

ALBERT EINSTEIN
SOCIEDADE BENEFICENTE ISRAELITA BRASILEIRA

kaggle

EXTRA/UNNECESSARY SLIDES

K Nearest Neighbors (KNN)

We used our training data to train the KNN model, using our K = 5:

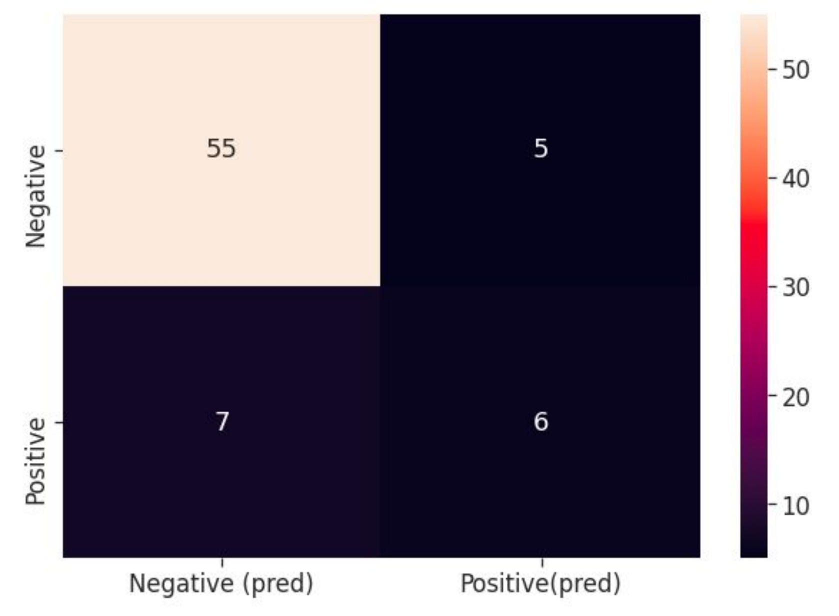
```
from          import
                                                     5
# Train the classifer

# Compute the score (mean accuracy) on test set

print 'KNN score: %f'
```

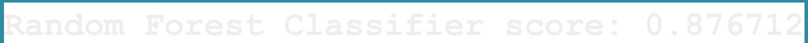
```
⇒ KNN score: 0.835616
```

Our Confusion Matrix can be seen here:



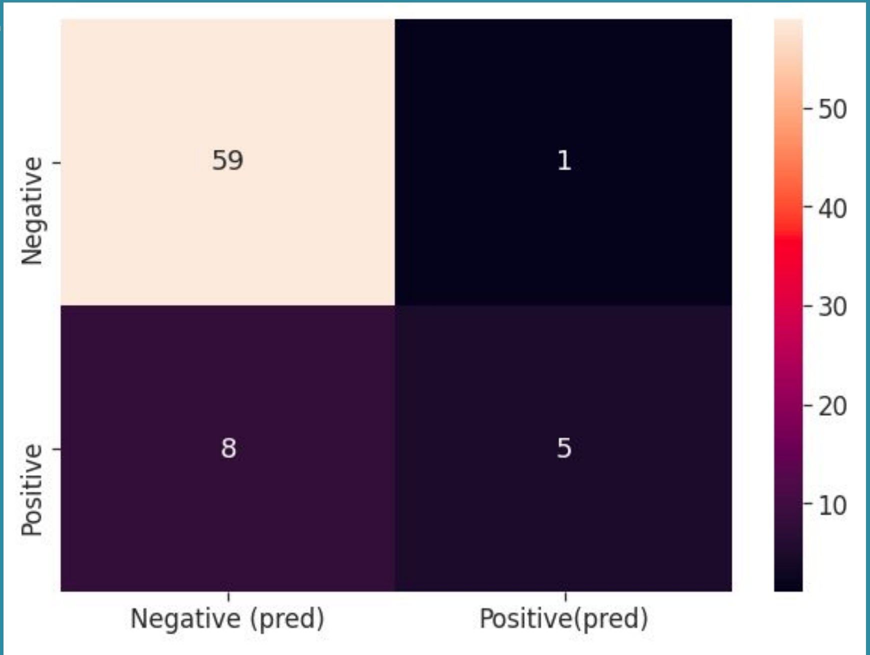
Random Forest Classifier

We did the Random Forest Classifier (RFC), and after trying out different values for n_estimators (which indicates the number of trees), we found n_estimators = 10 to be the most ideal value.



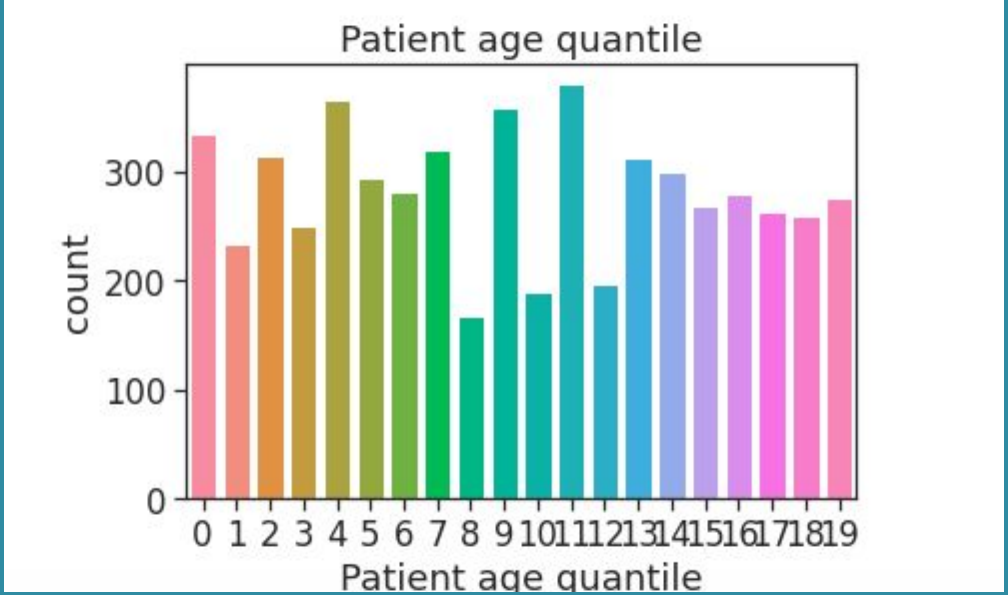
```
Random Forest Classifier score: 0.876712
```

RFC Confusion Matrix can be seen here:



Graphs - seaborn

Matplotlib plotted 23 individual bar plots, so we took measures to beautify our plots, using seaborn (sns):



cleaning our data

To add:

- Graph that shows the percentages of missing data (the different thresholds)
- First vs last category graphs
- The pathway (the process used)
- Distributions
    - Patient age quantile
    - Covid neg/pos !!

Creating our Machine Learning Model

We split our imported dataset (called nonan) into training and testing data:

```
from              _          import      _    _
 _        _       _        _             _    _                   4
                                                                 1
                                                                   1
                                               _     0.3       _      35
```

From here, we started using different classifiers to test the accuracy of our predicted data outcomes. Some of the classifiers we used include:

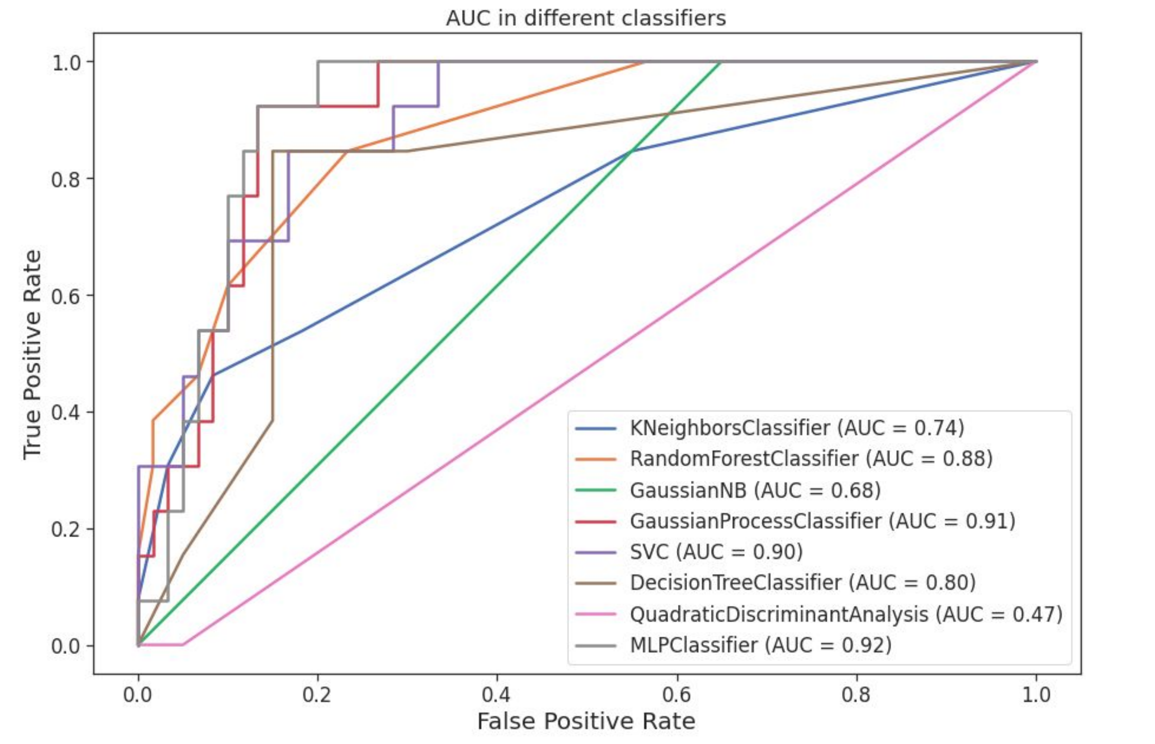
- K Nearest Neighbors
- Random Forest Classifier
- Naive Bayes
- Gaussian
- SVC
- Decision Trees

ROC/AUC Graphs

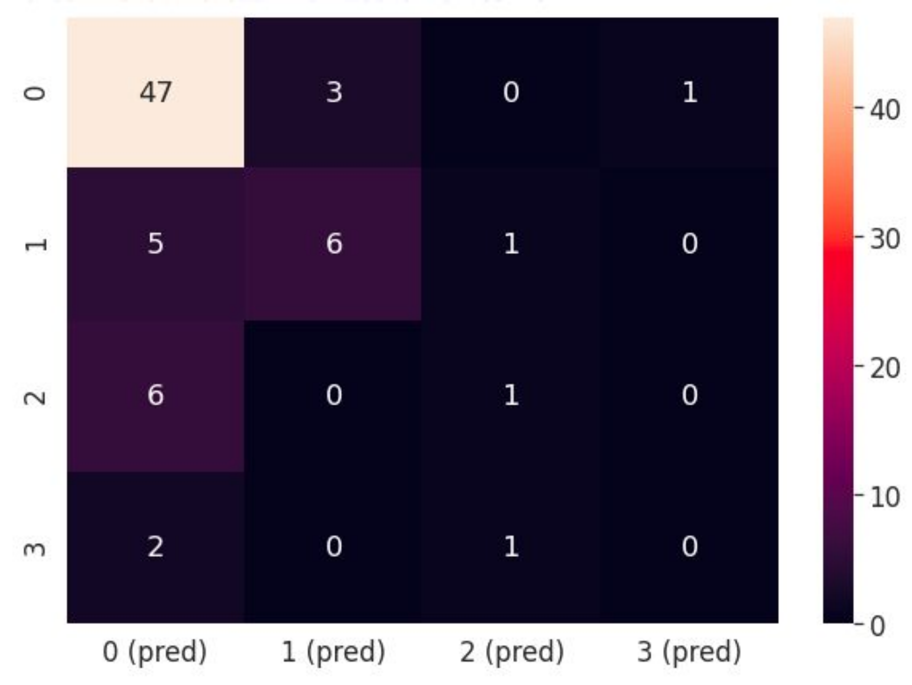
ROC: Receiver Operating Characteristic & **AUC:** Area Under Curve

The higher the area under the curve, the better the model’s performance is.

Ideally, the AUC should be above 0.5.



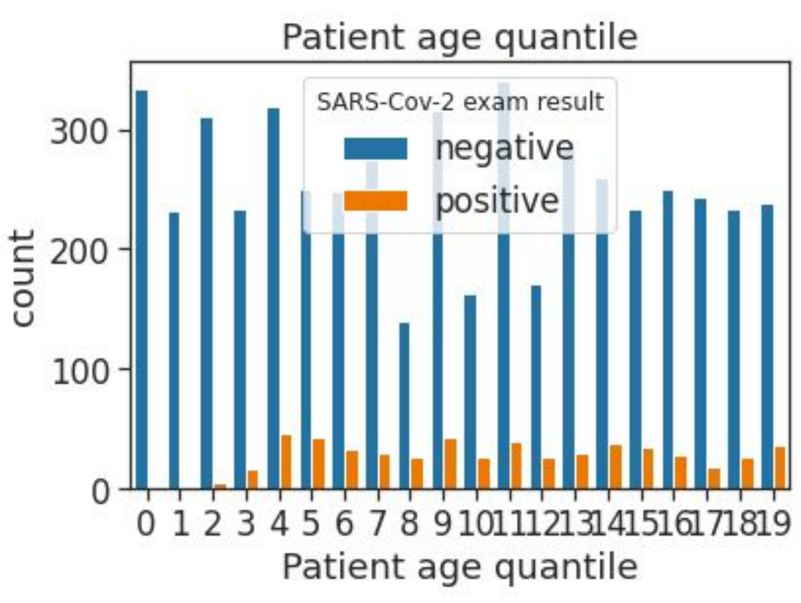
Random Forest for Severity

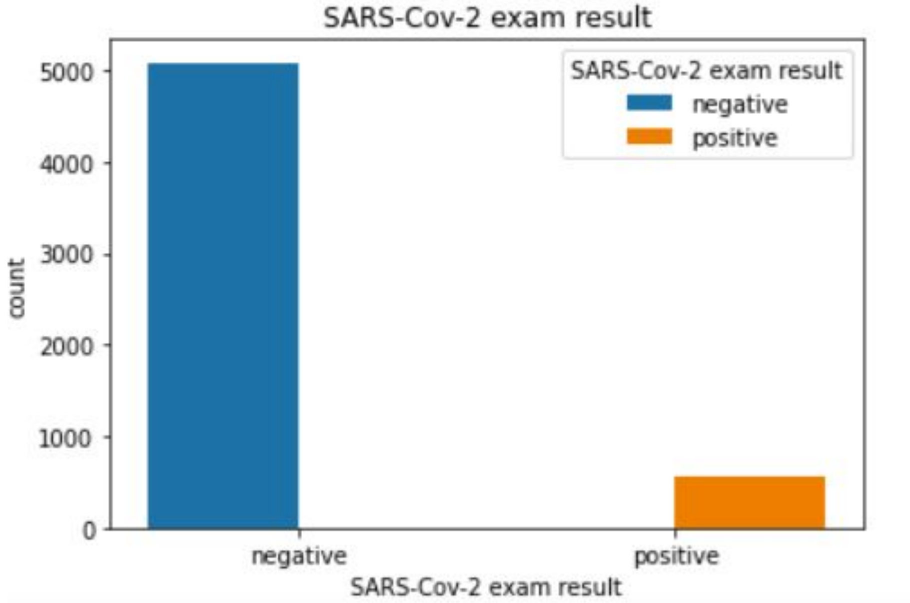


	0 (pred)	1 (pred)	2 (pred)	3 (pred)
0	47	3	0	1
1	5	6	1	0
2	6	0	1	0
3	2	0	1	0

Graphs - legends

So next, we plotted our graphs with a legend, specific to SARS-Cov-2 exam results: either positive or negative.





The Solution

On average % of people get tested for COVID.

- Using machine learning models, accurately + efficiently predict
    - COVID-19 patient status
    - COVID-19 patient severity
- Prevent further spread of COVID
- Help hospitals distribute beds + ICU spaces

Table of Contents (Task List)

1) Background (Gianna)
2) Data Cleaning (done)
3) Status Model (done)
   a) Comparing Accuracies (done)
   b) Confusion Matrices (done)
   c) ROC/AUC graph comparison (done)
   d) Our Choice (Jessica)
   e) False positive / negative consequences (done)
4) Severity Model (done)
   a) Comparing Accuracies (Vienna)
   b) Our Choice (Vienna)
   c) ROC/AUC graph (Vienna)
5) Bias and Limitations (Christina)
6) Important Features (Gianna)
   a) Decision Tree
7) Usage in Clinical Decision making (Srihita)
8) Conclusion (Fiona)
9) Acknowledgements

Presentation Slide Distribution

4) Gianna?

5) Gianna

6)

7) Christina + Srihita

8) Srihita

9) Srihita

10) Jessica

11) Jessica + Srihita

12) Jessica

13) Vienna

14) Vienna

15) Vienna + Christina

16) Vienna

17) Vienna

18) Christina

19) Jessica

20)Gianna

21) Gianna

22) Fiona

23) Fiona