Submitted by: Eden Dembinsky 212227888 & Assaf Lovton 2098444414 - HW1

Part 1- Data Loading and First Look:

Q1.

There are 3000 rows and 29 columns.

Q2.

```
2.0    754
1.0    707
3.0    554
0.0    400
4.0    272
5.0    120
6.0     29
7.0      7
8.0      2
9.0      1
Name: num_of_siblings, dtype: int64
```

This feature describes the number of siblings of a patient.

Q3.

The feature type should be ordinal. There is no reason for the feature to be continuous since the range is limited, so we need to choose between ordinal and categorical. If we choose ordinal we will be able to infer conclusions based on the number of siblings.

Feature name	Description	Type
patient_id	the unique identifier for a patient.	Continuous
age	the age of the patient.	Continuous
sex	the sex of the patient.	Categorial
weight	the weight of the patient.	Continuous
blood_type	the blood type: {A+, B-, etc'}	Categorial
address	the address of the patient	Other
current_location	coordinates of the current location of the patient on some grid	Other

job	the job of the patients	Other (There is a very large number of categories so we assume it is not)
num_of_siblings	the number of siblings of the patients.	Ordinal
happiness_score	A number descriving the amount of happiness the patient experiencing {-1..9}	Ordinal
household_income	the amount of money the patient's family makes.	Continuous
pcr_date	The data the PCR test was performed	Continuous
symptoms	describes the symptoms the patient has experienced	Other
sugar_levels	The sugar level of the patient	Continuous
sport_activity	The level of activity of patent {0..5}	Ordinal
conversations_per_day	The number of the people the person had contact with	Continuous
PCR_01	The result of the first feature of the PCR test	Continuous
PCR_02	The result of the second feature of the PCR test	Continuous
PCR_03	The result of the third feature of the PCR test	Continuous
PCR_04	The result of the fourth feature of the PCR test	Continuous
PCR_05	The result of the fifth feature of the PCR test	Continuous
PCR_06	The result of the sixth feature of the PCR test	Continuous
PCR_07	The result of the seventh feature of the PCR test	Continuous
PCR_08	The result of the eighth feature of the PCR test	Continuous
PCR_09	The result of the ninth feature of the PCR test	Continuous

PCR_10	The result of the tenth feature of the PCR test	Continuous

Q4.
The address type is other since there is no limit on the range of the values and there is no specific order that we can apply.
The current_location type is other since it is a tuple of coordinates, there is no specific order and no limit on the range of the values.

The job type is not categorial since the number of jobs is very large and we think that if we will add more data we will likely add new values to the optional categories. There is no specific order between the jobs therefore it is not continuous.

The num_of_sibilings type is ordinal. There is no reason for the feature to be continuous since the range is limited, so we need to choose between ordinal and categorical. If we choose ordinal we will be able to infer conclusions based on the number of siblings.

The happiness_score type is ordinal since there is a limited number of categories with a reasonable ordering, 9 is happier than 8…

The sport_activity type is ordinal since there is a limited number of categories with a reasonable ordering, 5 is more active than 3…

The symptoms type is not of type categorial since it is a list of categories for each patient and not a value that belongs to only one category. It is also not continuous since there is no specific order of the values.

Part 2- Data Imputation and Cleaning:

Q5.
We want to be able to reproduce the results we got every time we ran our model. If it will give different results every time it will be very hard to see if the changes we applied to the model contributed to its accuracy or if it has changed due to the different data used.

Univariate feature exploration:

Q6.
The length of the vector is 6.

Q7.
We added a one-hot encoding to the symptoms feature since it is hard to work with its current representation. We added it in the following format- categories: cough, fever, headache, low_appetite, shortness_of_breath therefore for a patient with symptoms of fever; headache we will get 0,1,1,0,0.
now we can easily apply sum/avg and more functions on each row and get the information while maintaining the relationship between the symptoms of each row (each patient).

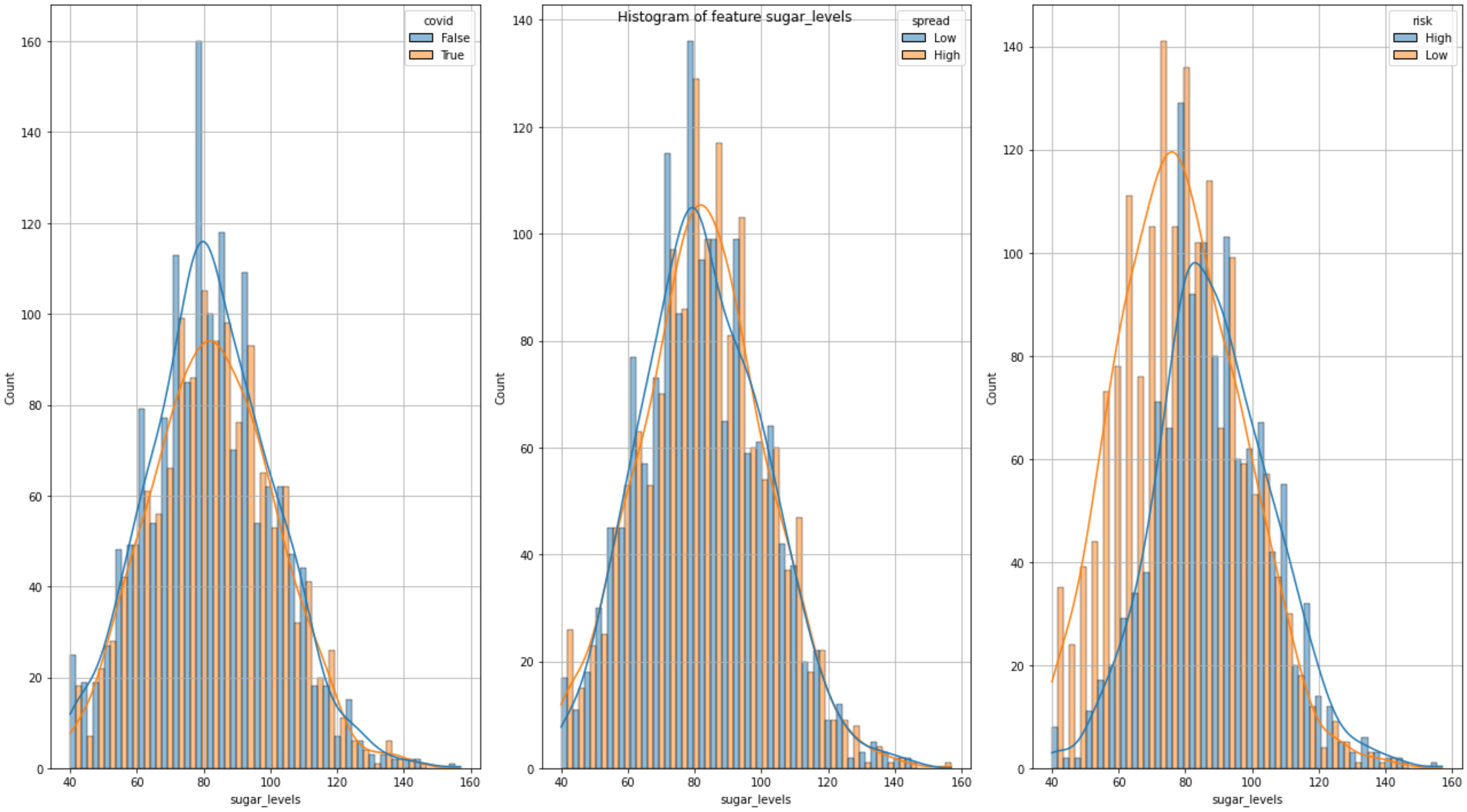
Q8.
We have already taken care of the symptoms-feature by creating the symptoms one hot with every possible symptom as a column feature (**symptom_cough, symotom_fever, symptoms_headache, symptoms_low_appetite, symptoms_shortness_of_breath, and symptoms_nan**).

We decided to take from the address feature the name of the state the patient lives at, as a new feature because we think that there is a connection between the state and the probability for covid and spread. So we create a **state_one_hot** feature of the states from the address feature- a column of every state. We were able to extract this information based on the observation that the state names are the seventh and eighth characters from the end.

We decided that the **job** feature will not be useful since the number of jobs is very big and the biggest group of a job has only 5 patients, therefore, we will lose this feature.

We also thought that it would be interesting to add a feature **housing_is_apt** describing if the patients live in apartments or a private house (we assumed that it is a private house if there is no Apt. in the address string).

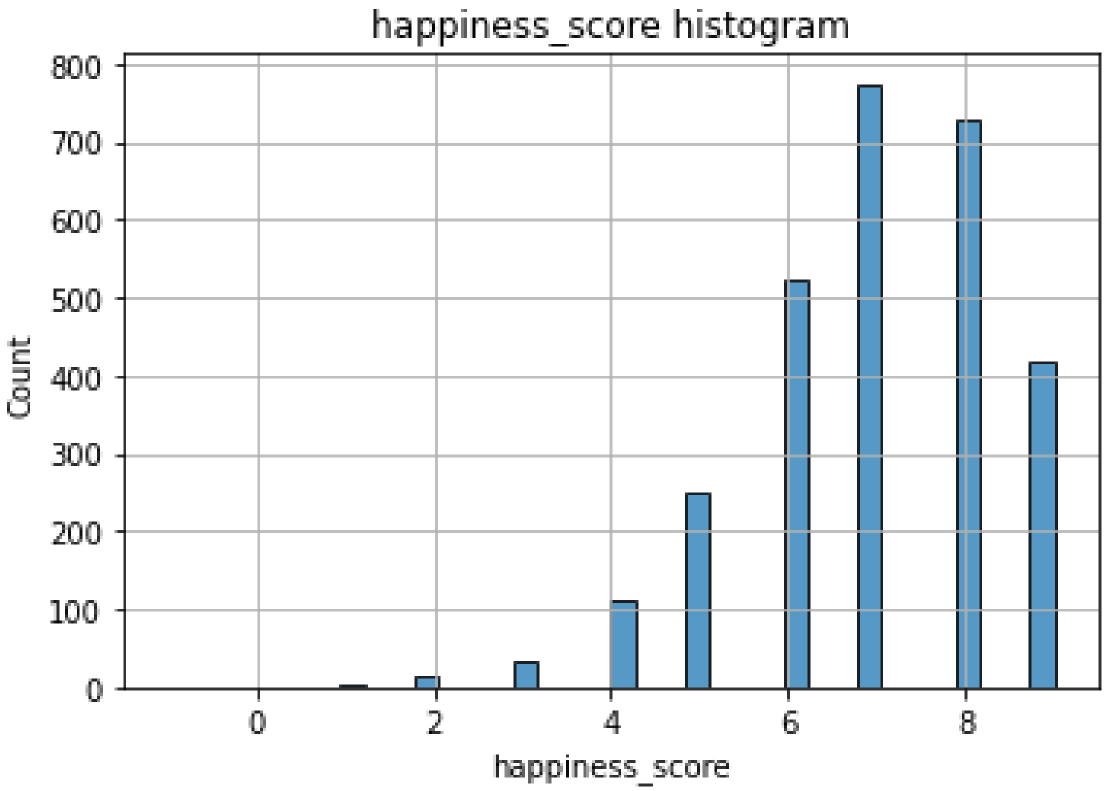
Q9.



As we can see we can only use sugar levels to estimate the risk. We can see in the rightmost graph that for low sugar levels we get a big amount of labels of low risk compared to the high risk. The difference becomes less significant from a sugar level equal to 80. For the spread and covid target features sadly we cannot use sugar levels since there is almost the same amount of both groups (false or true and low and high) across all sugar levels. Therefore we cannot find it as a useful feather for predicting these two target features.

Outlier Detection:

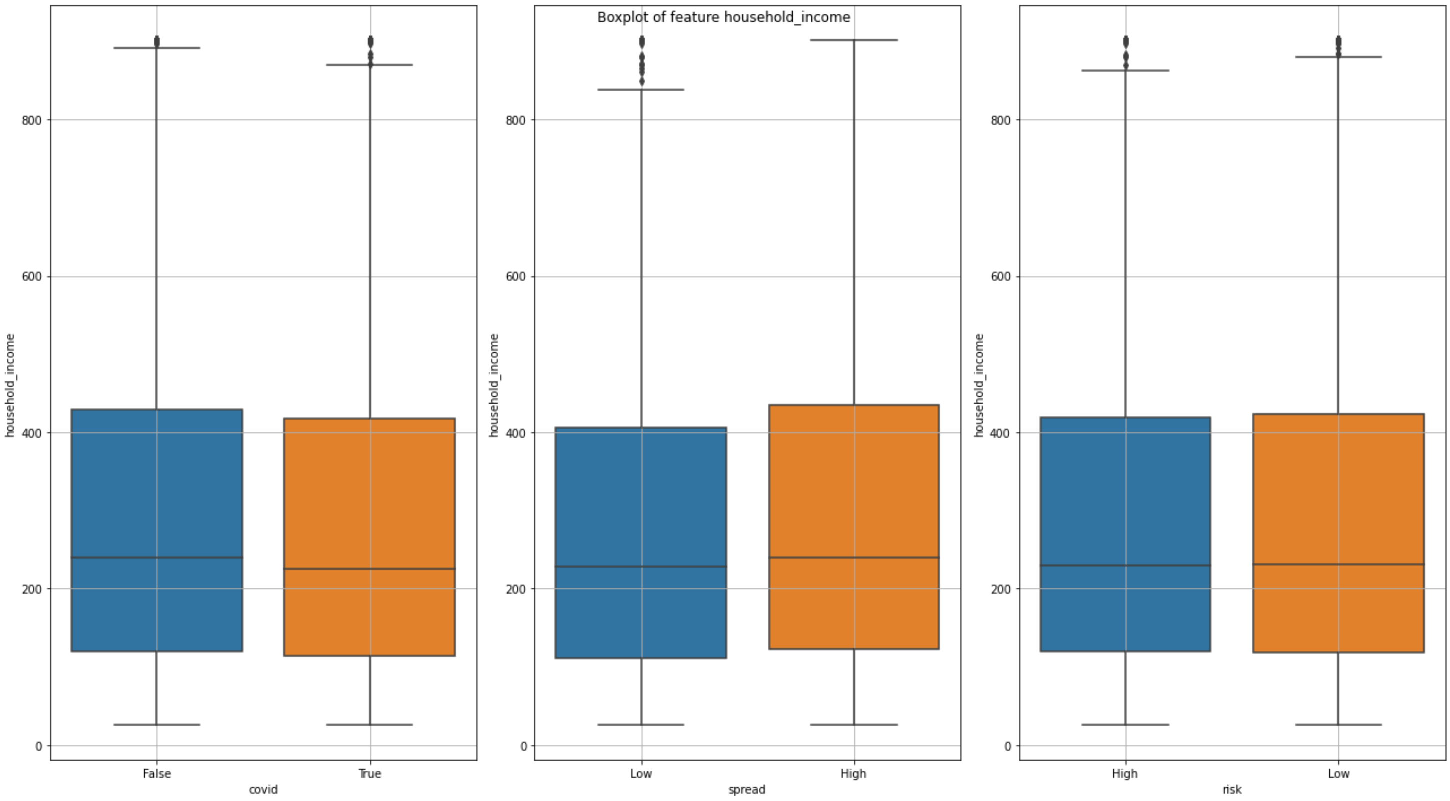
Q10.



```
7.0    774
 8.0    728
 6.0    523
 9.0    417
 5.0    250
 4.0    113
 3.0     35
 2.0     13
 1.0      2
-1.0      1
Name: happiness_score, dtype: int64
```

As we can see -1 is an outlier, we assume that the happiness score was measured between 1 to 9, and -1 is probably meant to say that there is no record of the score for this patient. We will take the trimming method and will get rid of this sample.

Q11.



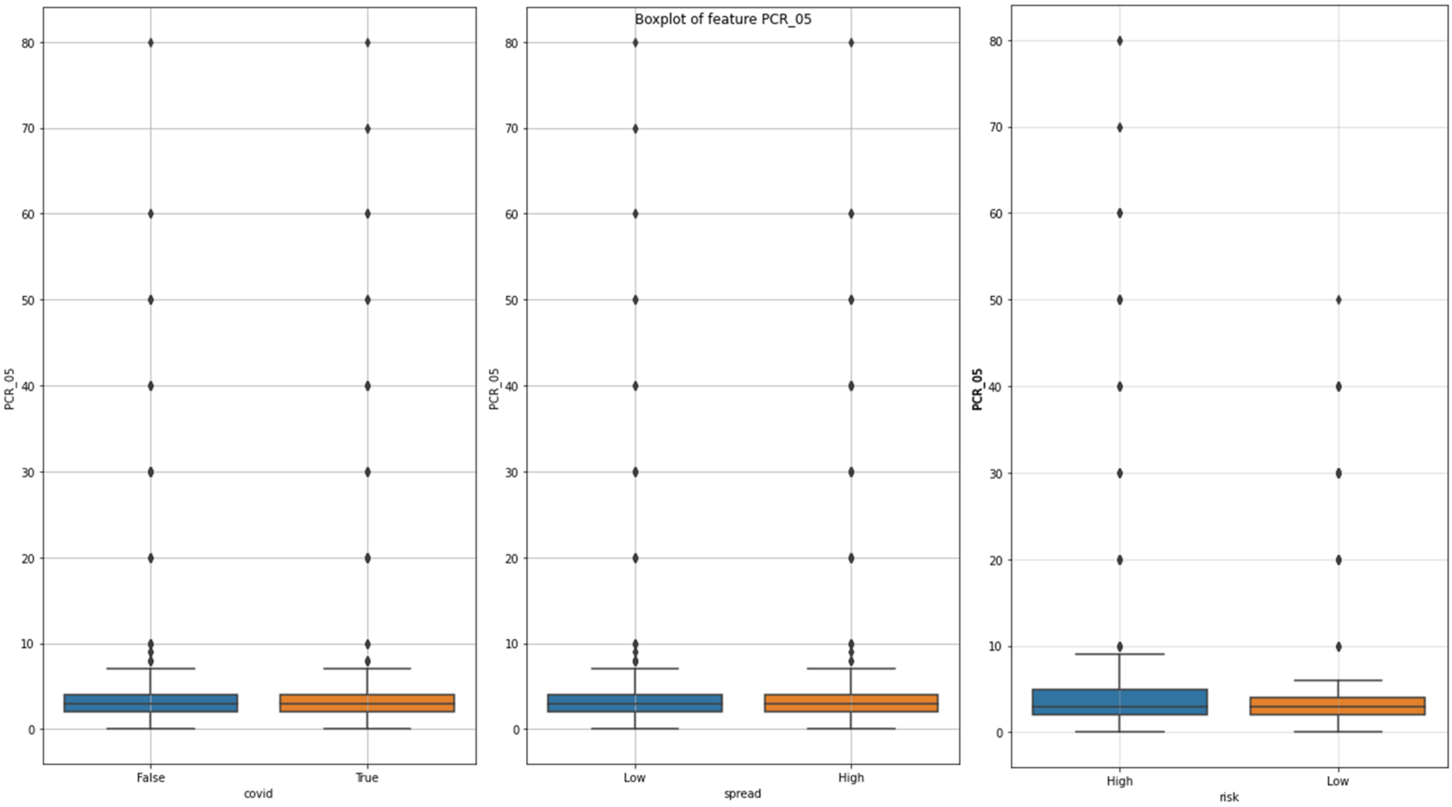
Since there are about 3000 samples, we cannot say that the number of outliers compared to the total number is significant enough to determine a threshold that will classify the risk with good confidence. We could set a threshold of 1750 that will classify as High risk, but since there are only 5 we cannot be sure whether it is a coincidence or a valid threshold.
Moreover, we can see that the household does not correlate with the risk even if we were to remove the outliers, and data of 3000 is not big enough to just erase data, any sample is important even if one of its features is an outlier.
We decided to go with the upper limit (<97%)and lower limit (>3%).
whereas if a sample is an outlier bigger than the upper limit we set his outlier sample to be the upper limit, this way we don't erase any data and we keep all our data in a range of 3 to 97 percent.

Q12.

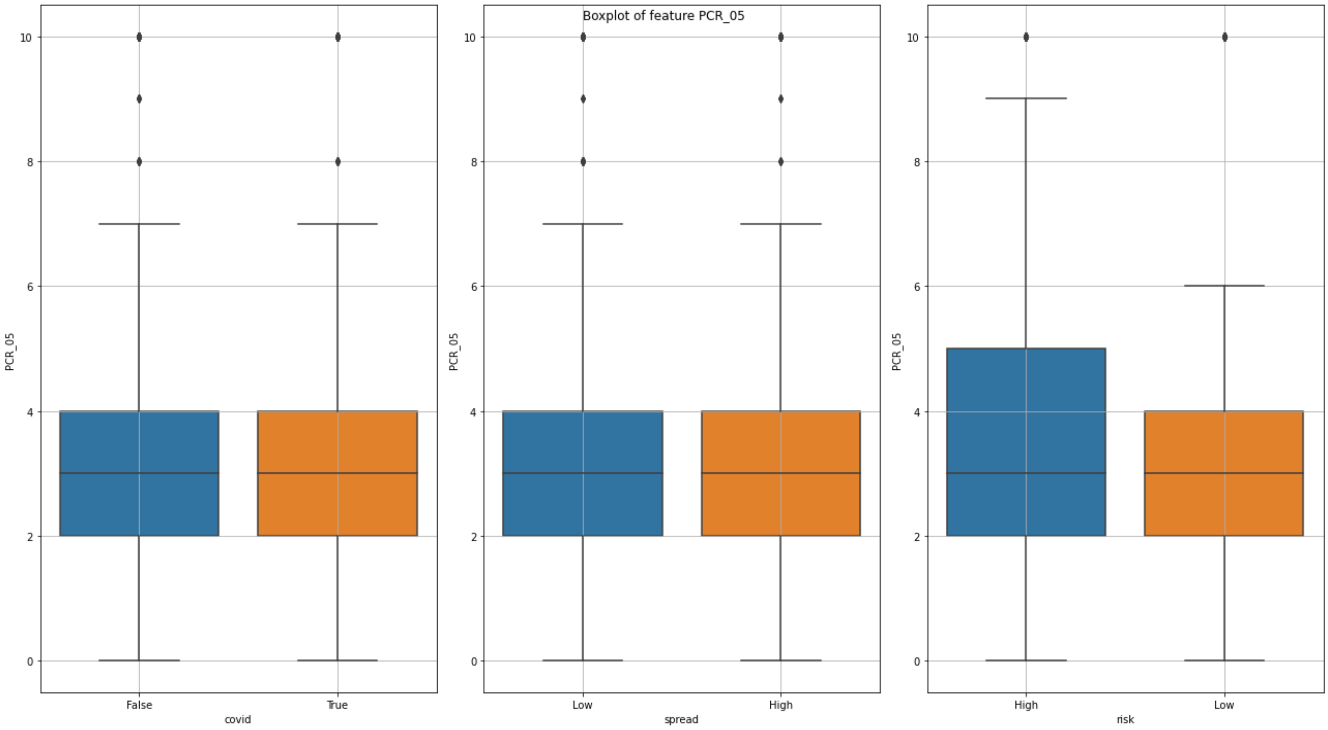
For starters, we used the describe function and looked at the min\max fields, we found out that there are no negative values for the weight and age features. It was hard to determine the correct range for the PCR features since we don't have information about what they are measuring. We did not want to drop the samples because they may have important values for the other features and because our dataset is pretty small.

In the beginning, we took the IQR method to deal with the outliers, but we got a negative lower limit since the 25% was smaller than 1.5* (IQR), therefore IQR did not handle the outliers, which made us understand that the assumption that these features are normally distributed was a wrong assumption. So we changed our method to work with an upper limit and a down limit. We tried many options and looking at the boxplots we decided to go with upper=97% and lower=3%. that provided us with almost no outliers in the box plots for the PCR features. look at the down figures.

Boxplot of feature PCR_05: Before handling outliers

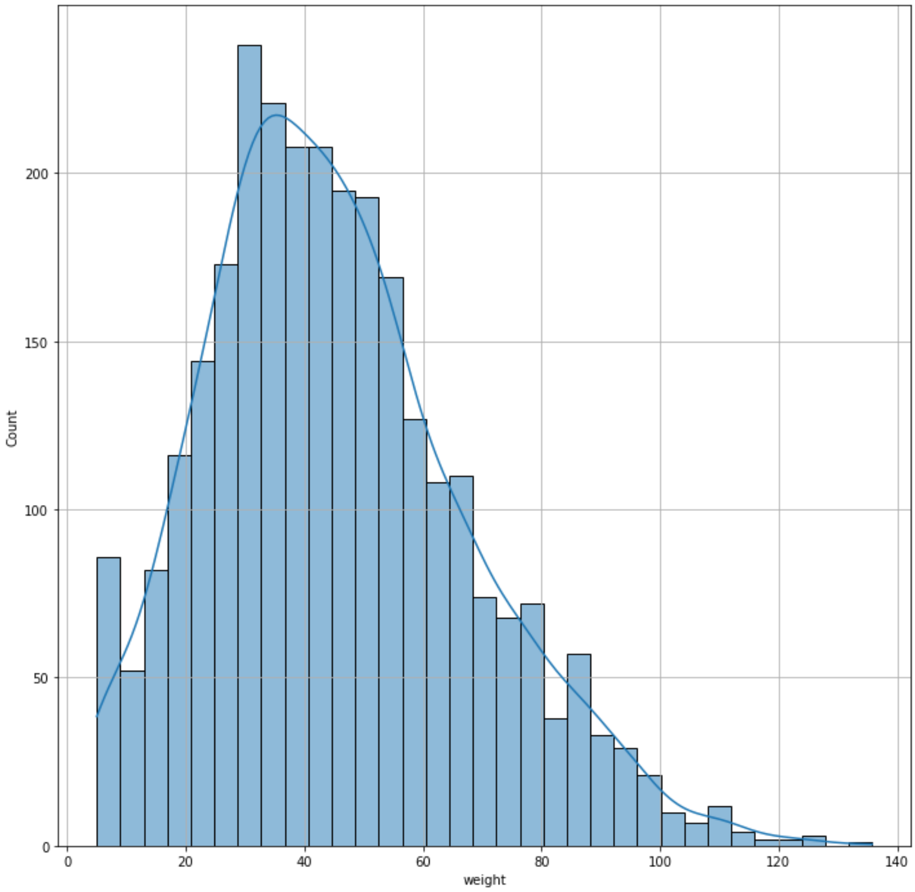


Boxplot of feature PCR_05: After handling outliers with the upper and lower limit method



On the other hand, after plotting the weight feature we found out that the distribution of this feature was quite good so we decided not to apply the upper-lower limit method since we don't want to lose valid data for no reason. The range of the values of wight stands in the normal weight range.

Histogram of Weight:



Missing data:

Q14.

One advantage of mean or median imputation is that it is easy and fast to implement and not add any outliers for example the boxplot won't change.
One disadvantage is that it distorts the original variable distribution and variance because assigning many samples to one value will change our distribution.

Q15.

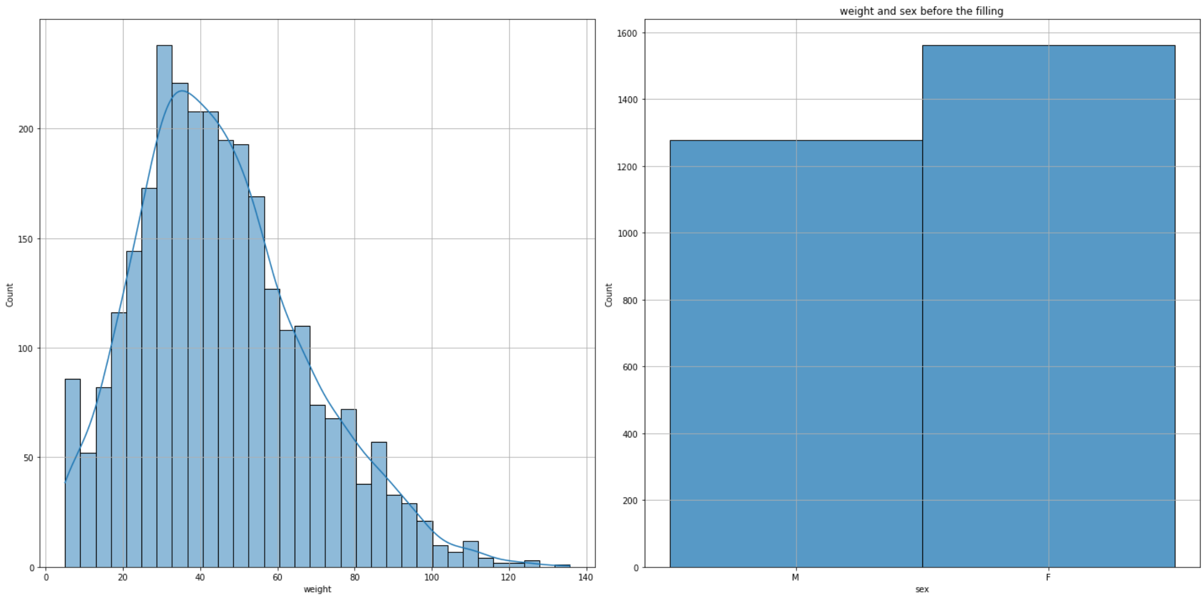
We don't mess with the other categories, we don't increase the most popular category and we keep the fact that we did not have the data, this might have a reason and significance and therefore can help us in the learning.

Q16.

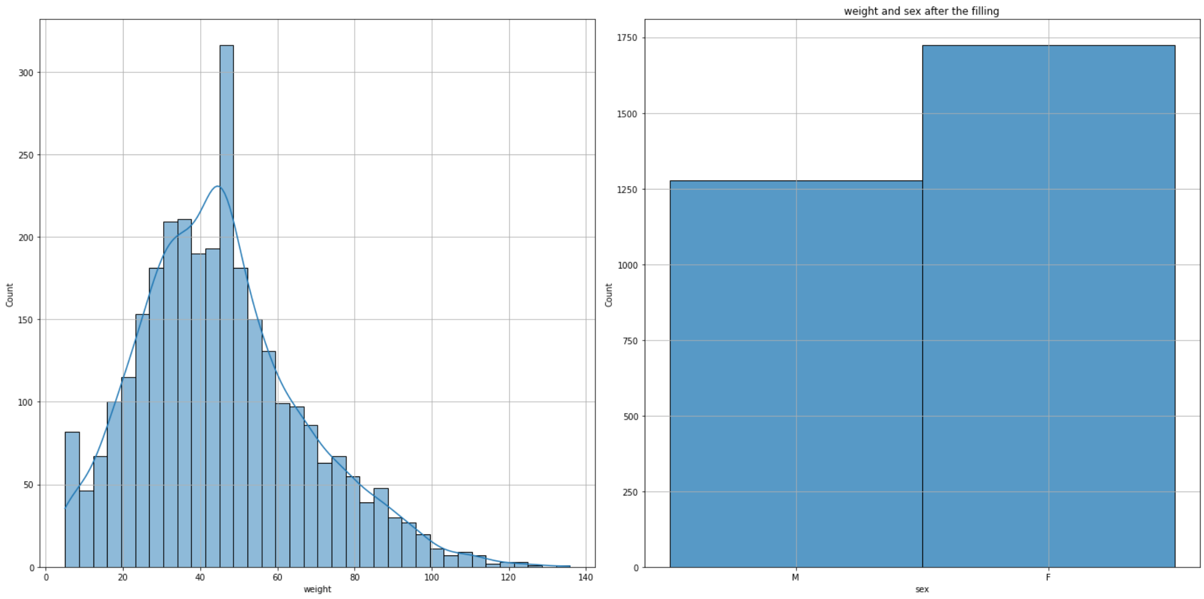
We decide to go with frequent category imputation since the data are missing at random and the missing observations most likely look like the majority of the observations Because it is most probable that there is no relation between the missing data and the number of siblings.

Q17.

Weight and sex before handling missing values:



Weight and sex after handling missing values:



Q18.

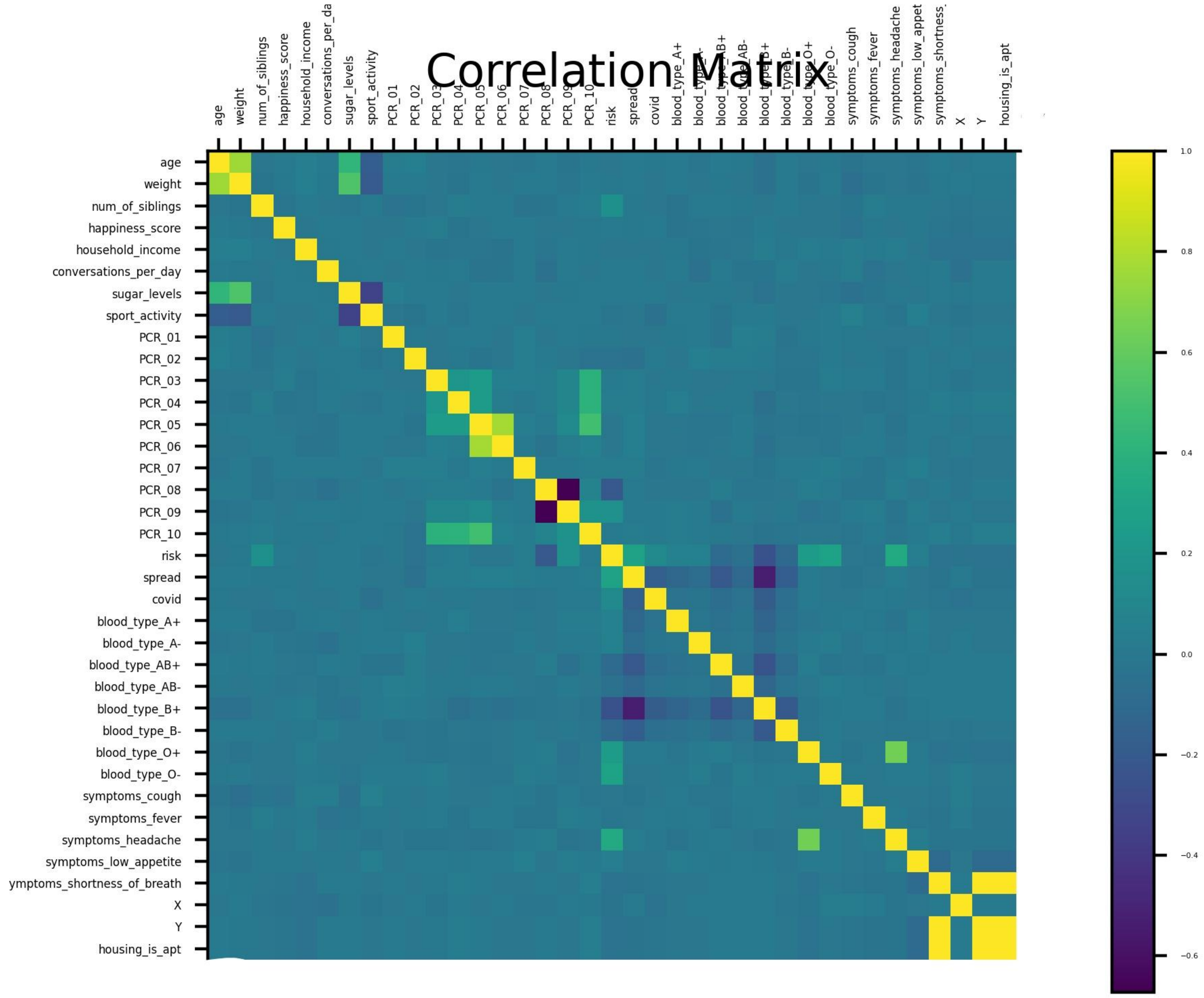
By handling outliers we are changing the data, changing the mean, changing the median, and such. So in order to have correct values to fill instead of the missing value, we need to first drop the outliers and only then fill the data with the mean/median of the data without the outliers.

If we used median or mean we might impute values that are calculated on the outlier causing them to be less accurate.

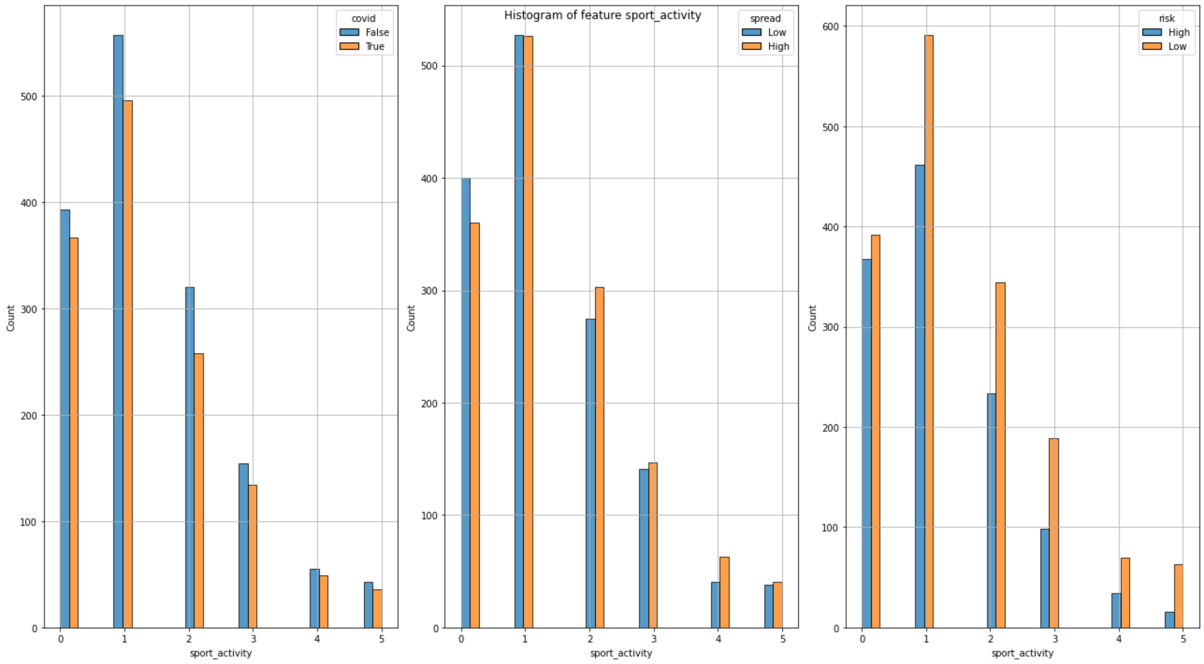
<u>Part 3: Feature Selection</u>

Q19.

After looking at the PCR_10 row we can see that according to the correlation with other continuous typed data, we found that the correlation between PCR_10 to PCR_03, PCR_04, PCR_05 thus, we can have one feature to represent all of them. Redundant features may hurt our learning performance.

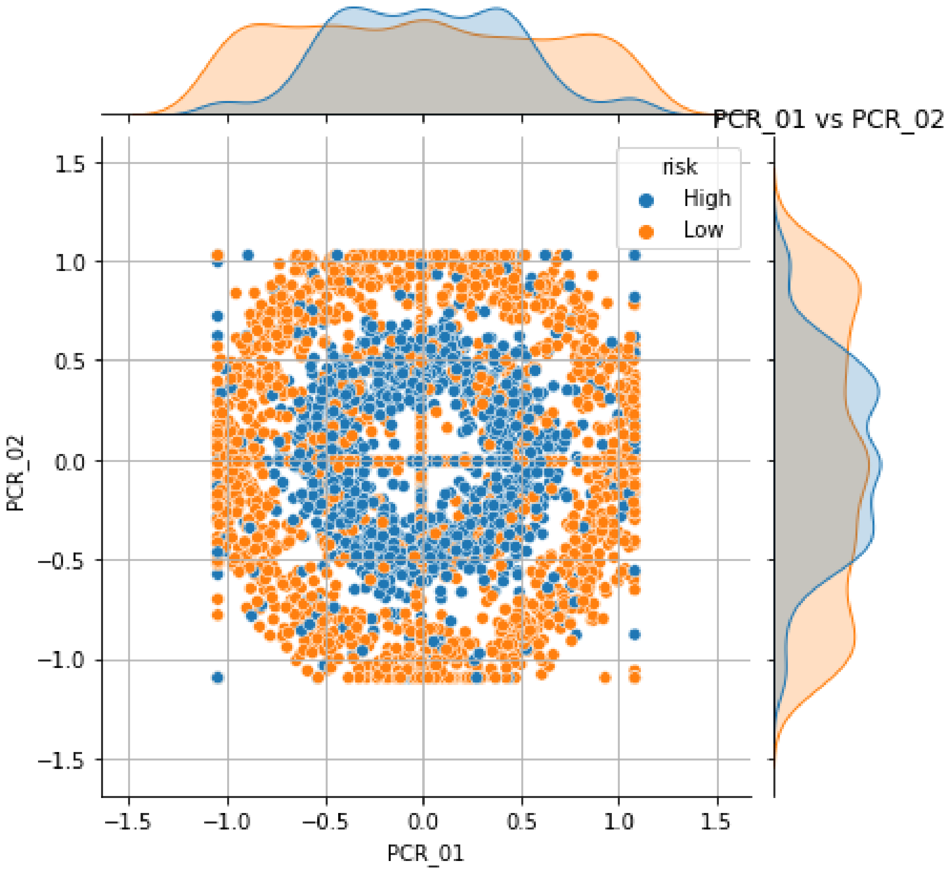


Q20.
No, this feature can’t help us with the target variable ‘covid’ because as we can see in the histogram the ratio between covid labels and no covid is pretty much the same for every sports activity score, so knowing that a patient has a certain sport activity score does not help us estimate if he has covid or not, on the other hand for the right graph corresponding to the risk target feature, we can see that for higher sports activity score there is a bigger chance that the patient has low risk, therefore the sports activity score can **not** help us estimate ‘covid’ but can help us estimate sports activity score.



Q21.

We can see in the graph that data is arranged in a circle with a center in (0,0). As we can see in the joint plot for a higher radius its more likely that the patient has lower risk and for a smaller radius he has higher risk, so we can apply a classifier that will classify the points in the inner circle as high and in outer circle as low, we can use a linear classifier with polar transformation as we saw in the lectures.



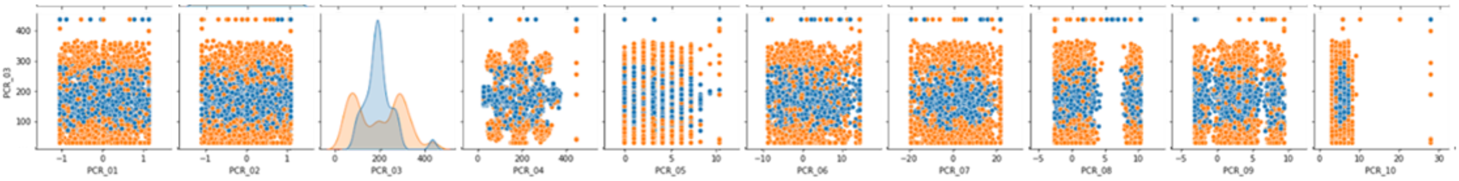
Q22.

- After observing the correlation matrix we can see that there is a strong correlation (~0.8) between PCR_05 and PCR_06.
- Moreover, as we saw in Q19, PCR_03, PCR_04, PCR_05, PCR_10 all have a similar correlation (~0.6) with each other.
- PCR_8 and PCR_9 (~ -0.85) have a strong negative correlation.
- We saw that age and weight look very similar in many of the pairs plots and also have a high correlation so we decided to get rid of the age feature.
- The sugar levels and sport activity features have a low negative correlation and as we saw in Q20 and Q9, both features can mainly help us estimate the target features-risk and no other target feature, thus we decided to keep only the sugar levels features.
- We also draw a huge three matrices of spread, risk, and covid against all features. They are too large to be shown inside this file, You can find them in the appendices section.

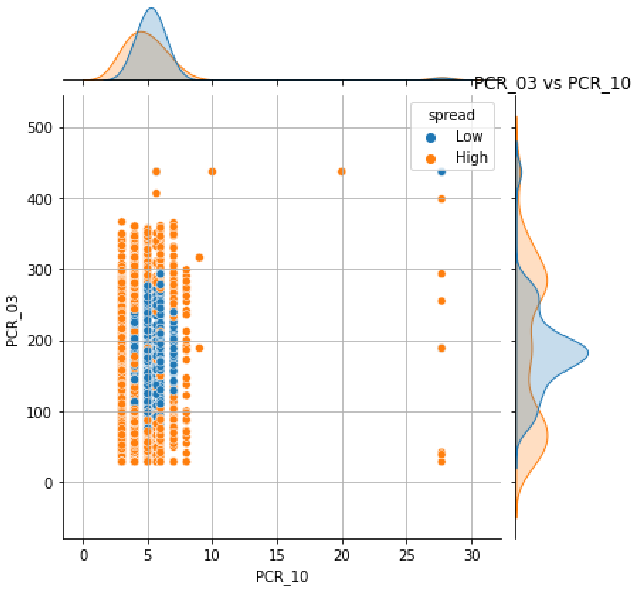
We plotted a pair plot between all PCR features, three times, every time for a different label (covid, risk, spread) in each pair plot we searched for some separation available between the orange points to the blue points as we saw in Q21.

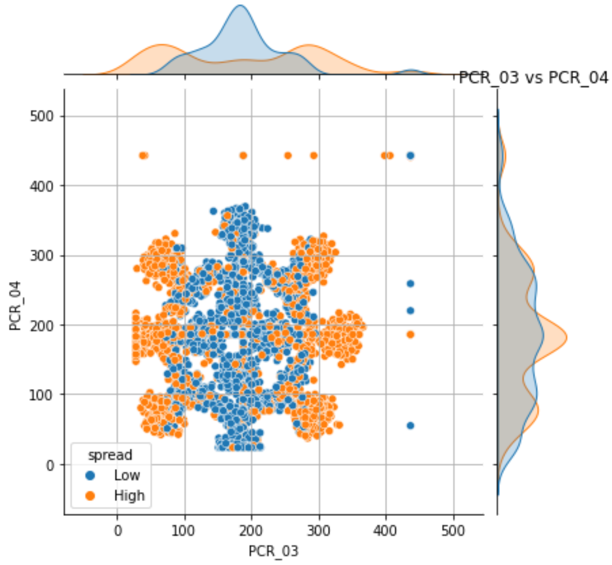
- After plotting a huge pair-plot of all PCR_features, we found out that PCR_03 separates the target feature-spread greatly. Here you can see the corresponding row of the matrix:

Pair plot of PCR_03 against all PCR features:

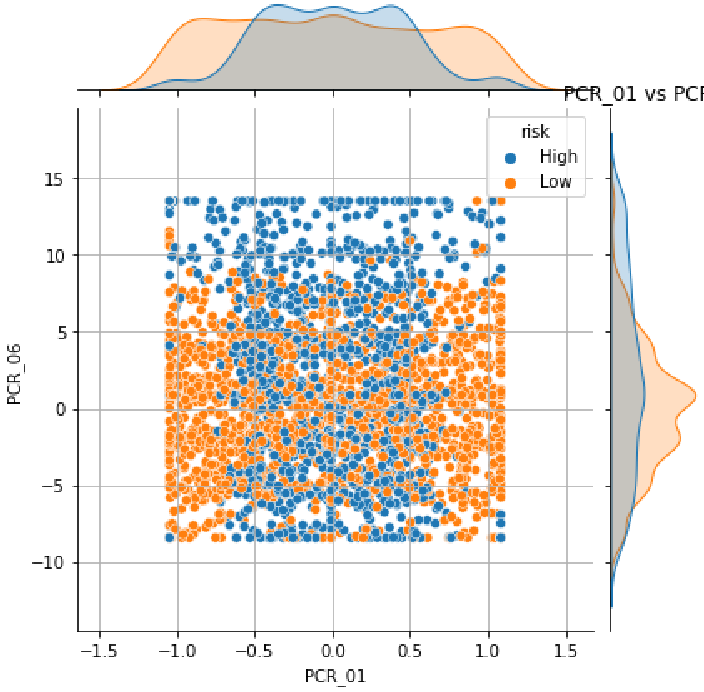


- As we can see the feature PCR_03 can separate pretty well with almost every PCR feature and especially with PCR_10 and PCR_04.

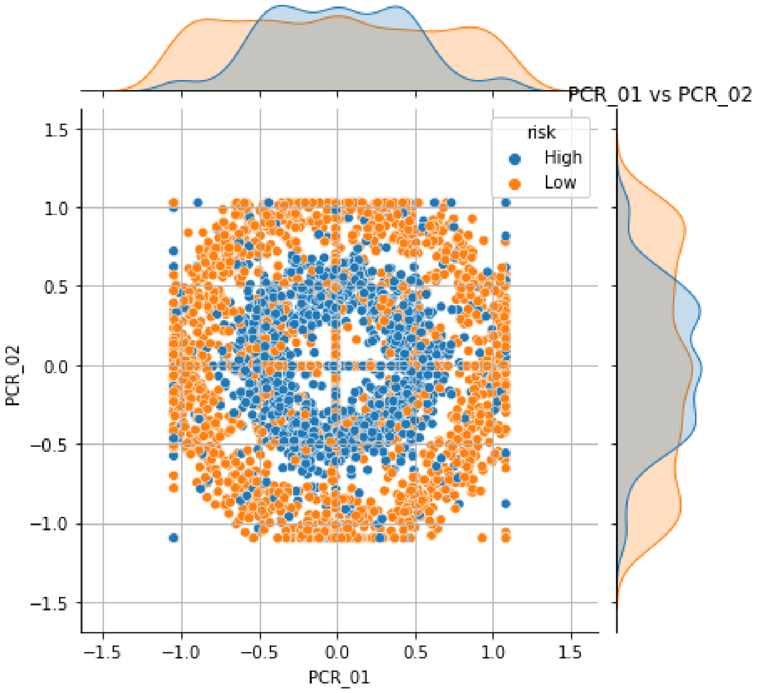




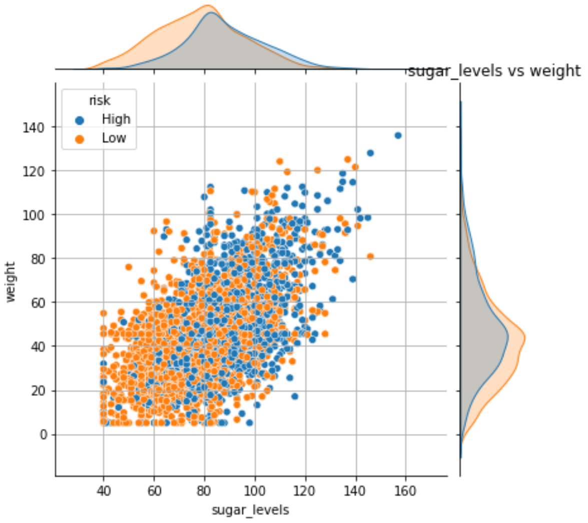
- We also found out that, PCR_06 is a good estimator for the target feature- risk, as we can see in the joint plot with PCR_01:

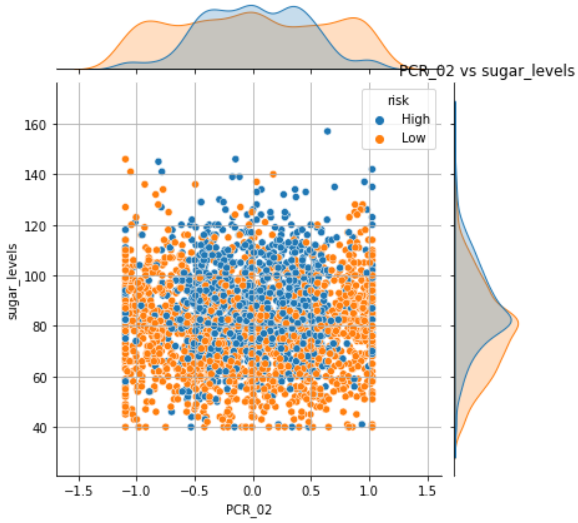


- Additionally, as we saw in Q21, PCR_01, and PCR_02 together can also help us estimate the risk label.

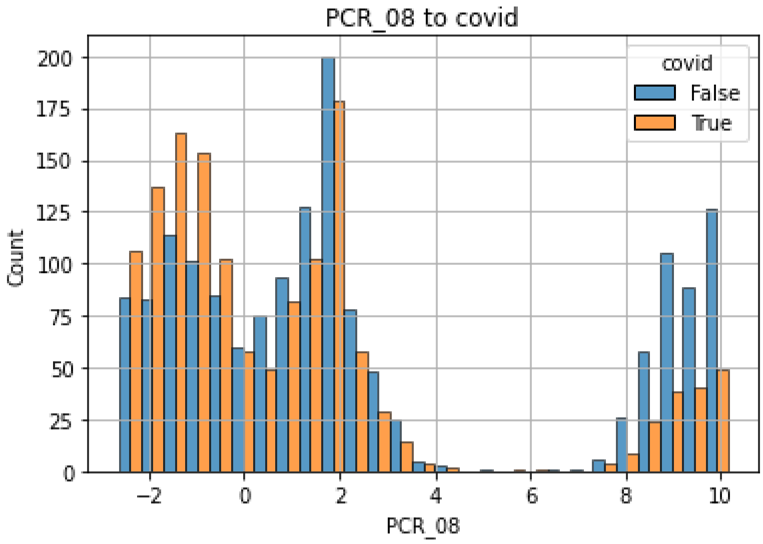


- Another feature that seems to help us with estimating the risk target feature is sugar_levels:

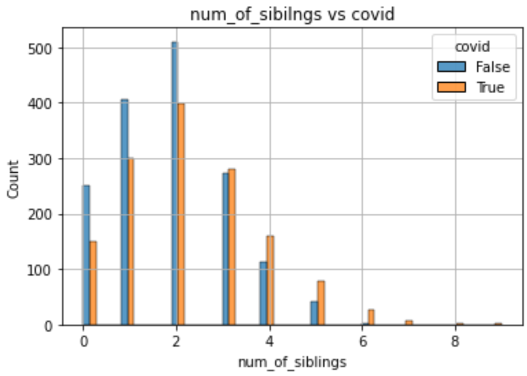




- We found PCR_08 helpful in estimating the false covid labels with some threshold on the higher PCR_08 values.

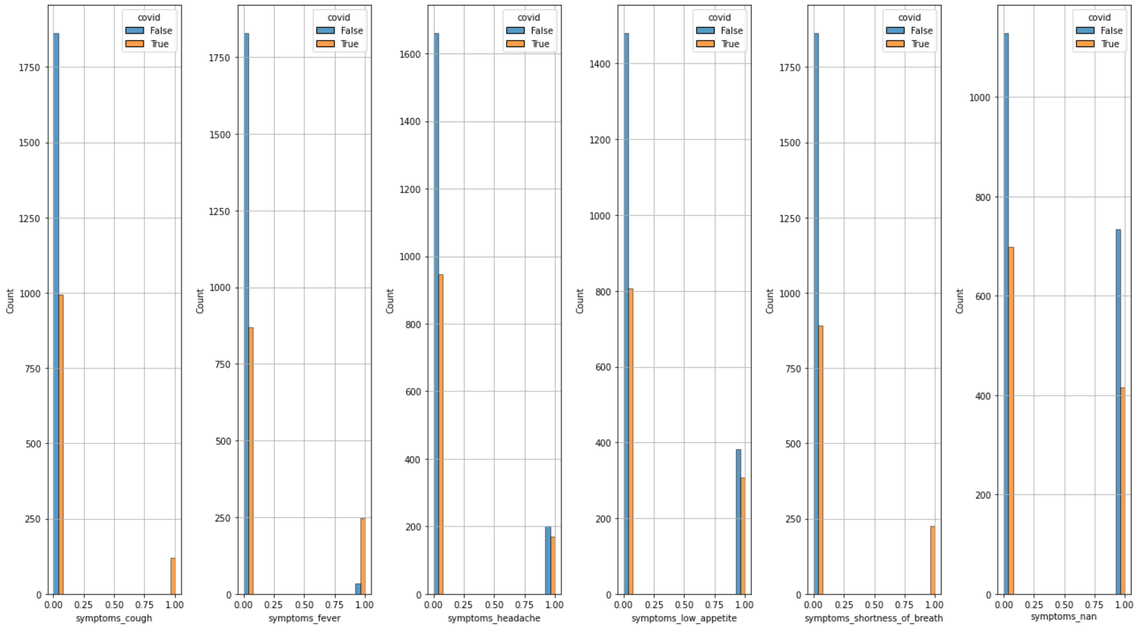


- We also ran pair plots between our other features (X, Y, conersations_per_day, etc..). We found that num_of_sibilngs is pretty useful to estimate covid:



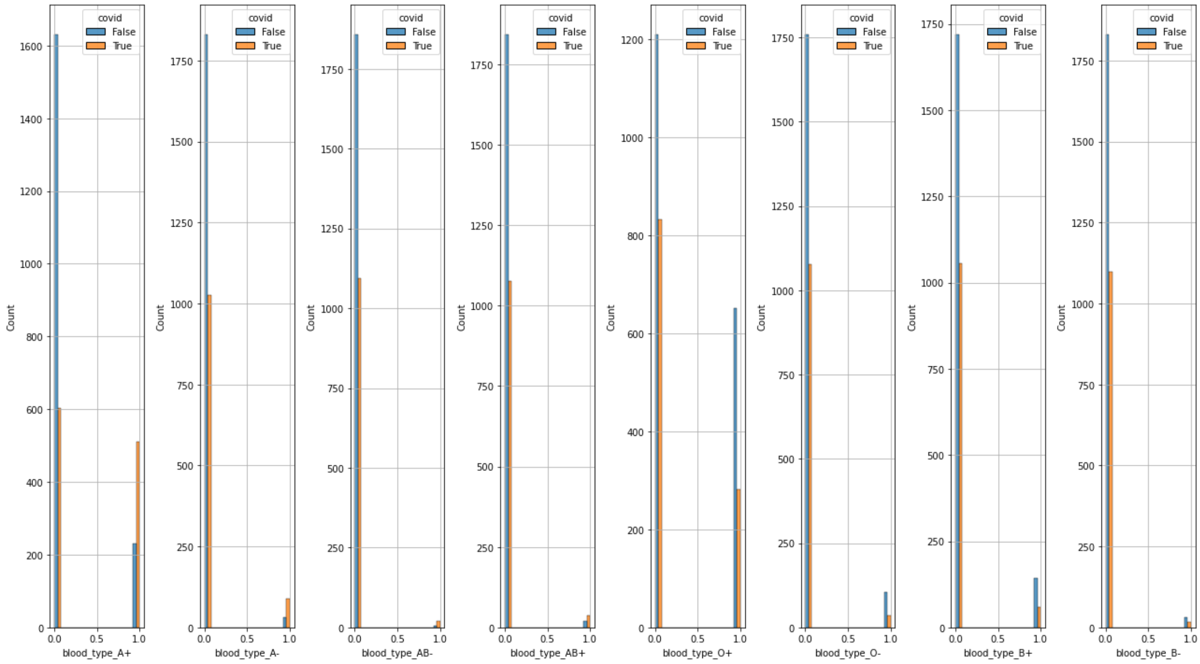
- We made 6 dummies features for each symptom that appears in our symptoms feature. We found that symptoms_cough, fever, shortness_of_breath are very useful to classify covid, thus we decided to keep them and drop the others.

Histogram of the symptoms against covid:



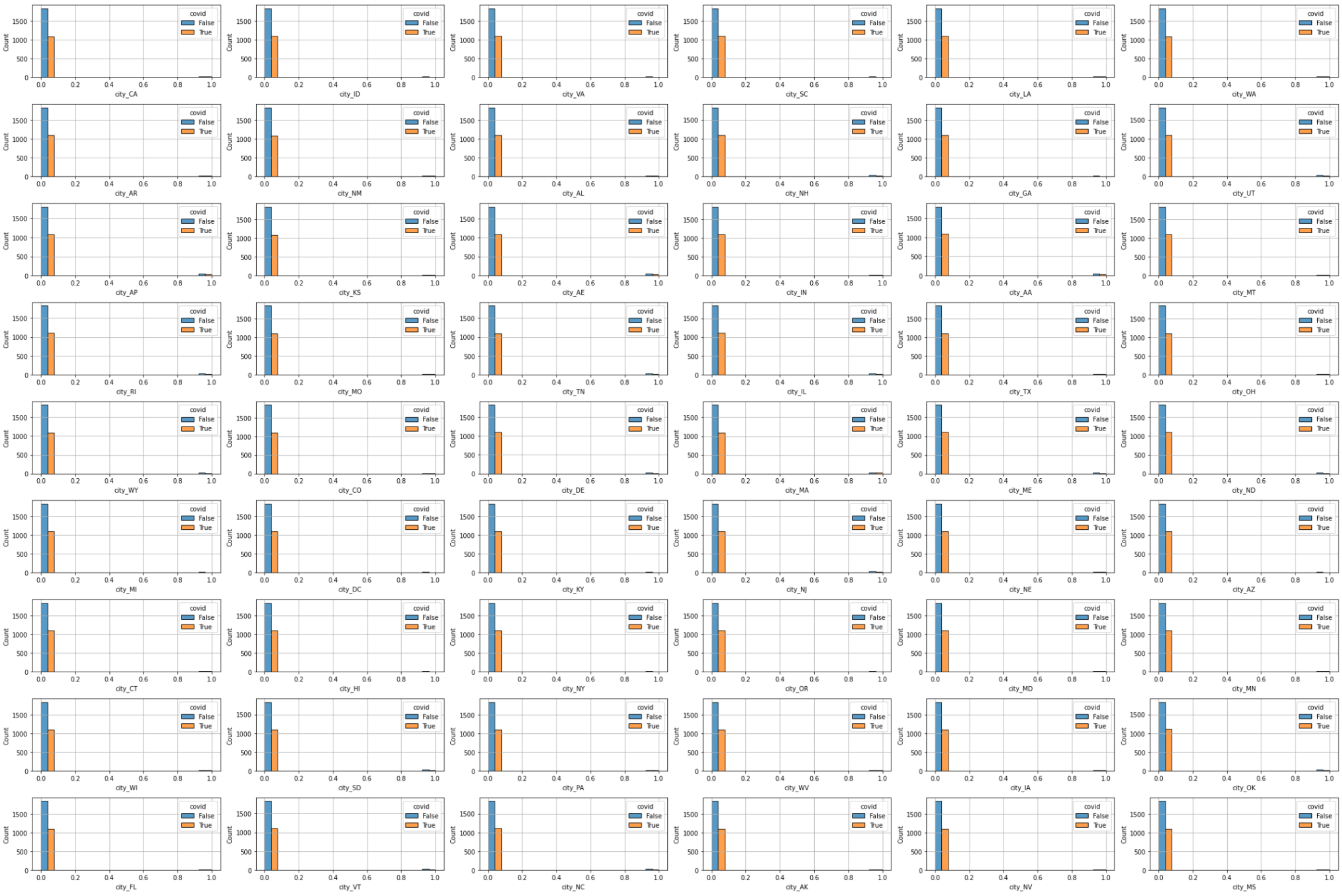
- As we can see the only significant feature is A+ since it can help estimate covid for true labels, because if A+ equals zero the probability for false covid is almost triple to probability for the true covid label, and if the A+ is one we get exactly the opposite, so knowing that a patient blood_type is A+ help us classify if has covid or not.

Histogram of all blood_types against covid target feature:



- As we can see there are not enough samples per state and moreover, they all distribute pretty much the same, for risk and spread the plot is similar, therefore we decided to drop the features of the states because we don't take much information from them on other features and for the target features.

histogram of all states against the target feature covid:



To conclude, after looking at many pair plots, joinplot, histplots, and searching for a pattern between numerous different features we decided to keep the following features: PCR_01, PCR_02,PCR_03, PCR_06, PCR_08, PCR_10, weight, sugar_levels, sex,num_of_sibilngs, household_income, blood_type_A+, symptoms_cough, symptoms_fever, symptoms_shortness_of_breath.
And of course, the target features risk, covid, and spread.
These features help us the most to identify the target features as we explained above.

Q23.

Feature name	Keep	New	Explanation
patient_id	X	X	Does not provide us with important information, it is good for searching the database not for training our model
age	X	X	Highly correlated with weight, no extra information.
sex	V	X	sex is important with the weight feature: a woman with a weight of 80, should be treated differently to a guy who weighed 80.
weight	V	X	An important feature helps us estimate risk as we saw above.
blood_type	X	X	Turned into a one-hot vector of all different types of blood.
blood_type_B+	X	V	We saw in the plot above that this feature does not help us classify any target feature.
blood_type_B-	X	V	We saw in the plot above that this feature does not help us classify any target feature.
blood_type_O+	X	V	We saw in the plot above that this feature does not help us classify any target feature.
blood_type_O-	X	V	We saw in the plot above that this feature does not help us classify any target feature.
blood_type_AB+	X	V	We saw in the plot above that this feature does not help us classify any target feature.
blood_type_AB-	X	V	We saw in the plot above that this feature does not help us classify any target feature.
blood_type_A-	X	V	We saw in the plot above that this feature does not help us classify any target feature.
blood_type_A+	V	V	We found that this feature helps us estimate the covid feature, in the plot above.
happiness_score	X	X	We could not see any relation between this feature to any other features in the pair plots we drew.
household_income	V	X	we found that this feature is pretty useful especially with other features as well.
job	X	X	This feature is redundant since there are a lot of jobs and each job does not give us any important information.

current_location	X	X	We drop this feature and instead created 2 new features: X, Y
X(current_location)	X	V	We thought that these features with the combination of the Y feature will help us a lot, unfortunately, we didn't find any interesting relationship with this feature to others.
Y(current_location)	X	V	same as X
is_housing_aprtment	X	V	We created this feature from the address feature’ after looking at plots we didn't see any unique relation to other features.
state_X where every X represents a name of a state	X	V	After plotting the data we noticed that there are not enough samples per state and moreover they all distribute pretty much the same.
pcr_date	X	X	we didn't find this feature interesting to create from his other features.
symptoms	X	X	We created a one-hot vector feature for every symptom and removed this feature-you can see them down below
symptoms_cough	V	V	We found out that cough is a good estimator for covid since there were no False covid labeled people that reported this symptom. in contrast to above 100 Ture covid labels that reported this symptom.
symptoms_fever	V	V	We found out that cough is a good estimator for covid since that patients that reported on this symptom are much more probable to have covid, as we saw in the plot above.
symptoms_headache	X	V	We did not find this feature a good estimator for any of the target features, since we got pretty equal results for the patient that reported this symptom and for patients that did not.
symptoms_low_appetite	X	V	We did not find this feature a good estimator for any of the target features, since we got pretty equal results for the patient that reported this symptom and for patients that did not.
symptoms_shortness_of_breath	V	V	We found out that this feature is a good estimator for covid since there were no False covid labeled people that reported this symptom. in contrast to almost 250 Ture covid labels that reported this symptom.

PCR_01	V	X	We found it important when it combined with PCR_02 to estimate the risk, as we saw in Q21.
PCR_02	V	X	The same reason as PCR_01.
PCR_03	V	X	We found it a good estimator for spread target feature when paired plotted with every other PCR feature, especially with PCR_10.
PCR_04	X	X	This feature is correlated with PCR_10 and does not add any useful information beyond that.
PCR_05	X	X	This feature is correlated with PCR_10 and does not add any useful information beyond that.
PCR_06	V	X	We saw that PCR_06 combined with PCR_01 can estimate the risk target feature pretty well.
PCR_07	X	X	We didn't find any relation between this feature to other features.
PCR_08	V	X	We found a strong relationship between this feature to covid as we saw in the histplot above.
PCR_09	X	X	This feature has a strong negative correlation with PCR_08, so we decided to keep PCR_08 instead.
PCR_10	V	X	This feature help PCR_03 estimate spread, as we saw in the plot above.
conversation_per_day	X	X	We didn't find any useful relations between this feature to others.

Appendices:

a.
Pairplot of covid against all features
https://drive.google.com/file/d/1V0tg5jMJEhMldx9h0SAFMp0Lei7vZAoP/view?usp=sharing

b.
Pairplot of risk against all features
https://drive.google.com/file/d/1LIUmNG1iEX2y6BPmDEhJCph0MQ-U5kRo/view?usp=sharing

c.
Pairplot of spread against all features
https://drive.google.com/file/d/1KGJQClCUZfZYjkjOynYCxafTHbAxpW9B/view?usp=sharing