Chronic Kidney Disease Classification

##### **Team Members**

Abdulrahman Saud Al Saleh - [218110112@psu.edu.sa](mailto:218110112@psu.edu.sa)  
Yousef Al Asfar - 216210574@psu.edu.sa

##### **Resources**

1. Chronic KIdney Disease Dataset <https://www.kaggle.com/mansoordaku/ckdisease>

# **Introduction**

In this project, we are given a dataset containing patients’ health records and whether or not they are diagnosed with chronic kidney disease or not. Our objective is study the relationship between the features and program an algorithm that can help classify patients who have the disease or not.

# **Background**

Chronic kidney disease the gradual loss of kidney function. In the early stages, you might have few signs or symptoms. You might not realize that you have kidney disease until the condition is advanced. The kidneys filter wastes and excess fluids from your blood; chronic kidney disease can cause dangerous levels of fluid, electrolytes and wastes to build up in your body.

**Symptoms**:

* Nausea
* Vomiting
* Loss of appetite
* Fatigue and weakness
* Sleep problems
* Urinating more or less
* Decreased mental sharpness
* Muscle cramps
* Swelling of feet and ankles
* Dry, itchy skin
* High blood pressure (hypertension)
* Shortness of breath, if fluid builds up in the lungs
* Chest pain
* Type 1 or type 2 diabetes
* High blood pressure
* Interstitial nephritis
* Polycystic kidney disease
* inherited kidney diseases
* Prolonged obstruction of the urinary tract
* Vesicoureteral reflux
* Pyelonephritis

**Risk Factors**

* Diabetes
* High blood pressure
* Heart (cardiovascular) disease
* Smoking
* Obesity
* Being Black, Native American or Asian American
* Family history of kidney disease
* Abnormal kidney structure
* Older age
* Frequent use of medications that can damage the kidneys

Considering all of this, we have to assume all the features provided (besides ID) are relevant for our classification because they can influence all of the following factors and symptoms mentioned above.

# **Data Exploration**

The data was taken over a 2-month period in India with 25 features (red blood cell count, white blood cell count, etc). The target is the 'classification', which is either 'ckd' or 'notckd' (has chronic kidney disease or does not). There are 400 rows

**Attribute Information:**

We use 24 + class = 25 (11 numeric ,14 nominal)

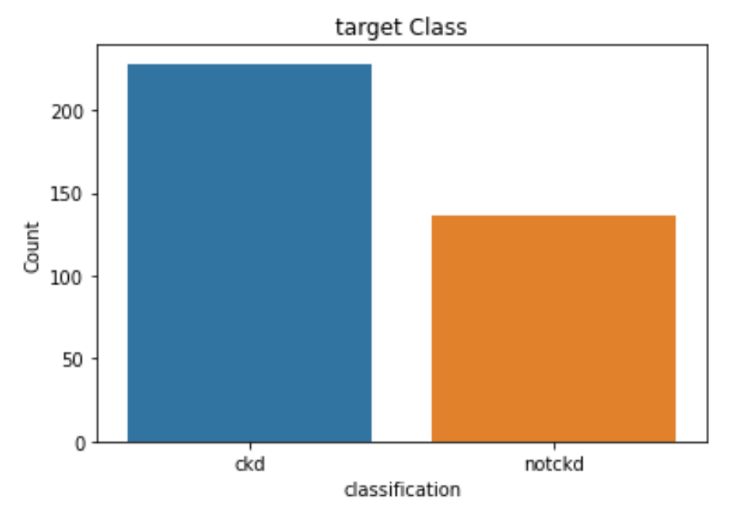
1. Age(numerical)   
   age in years
2. Blood Pressure(numerical)   
   bp in mm/Hg
3. Specific Gravity(nominal)   
   sg - (1.005,1.010,1.015,1.020,1.025)
4. Albumin(nominal)   
   al - (0,1,2,3,4,5)
5. Sugar(nominal)   
   su - (0,1,2,3,4,5)
6. Red Blood Cells(nominal)   
   rbc - (normal,abnormal)
7. Pus Cell (nominal)   
   pc - (normal,abnormal)
8. Pus Cell clumps(nominal)   
   pcc - (present,notpresent)
9. Bacteria(nominal)   
   ba - (present,notpresent)
10. Blood Glucose Random(numerical)   
    bgr in mgs/dl
11. Blood Urea(numerical)   
    bu in mgs/dl
12. Serum Creatinine(numerical)   
    sc in mgs/dl
13. Sodium(numerical)   
    sod in mEq/L
14. Potassium(numerical)   
    pot in mEq/L
15. Hemoglobin(numerical)   
    hemo in gms
16. Packed Cell Volume(numerical)  
    pcv
17. White Blood Cell Count(numerical)   
    wc in cells/cumm
18. Red Blood Cell Count(numerical)   
    rc in millions/cmm
19. Hypertension(nominal)   
    htn - (yes,no)
20. Diabetes Mellitus(nominal)   
    dm - (yes,no)
21. Coronary Artery Disease(nominal)   
    cad - (yes,no)
22. Appetite(nominal)   
    appet - (good,poor)
23. Pedal Edema(nominal)   
    pe - (yes,no)
24. Anemia(nominal)   
    ane - (yes,no)
25. Class (nominal)   
    class - (ckd,notckd)

**After importing the dataset, we check its unique values, and we find a bunch of values containing “/t?” and such, we need to clean this dataset before we use it any further**

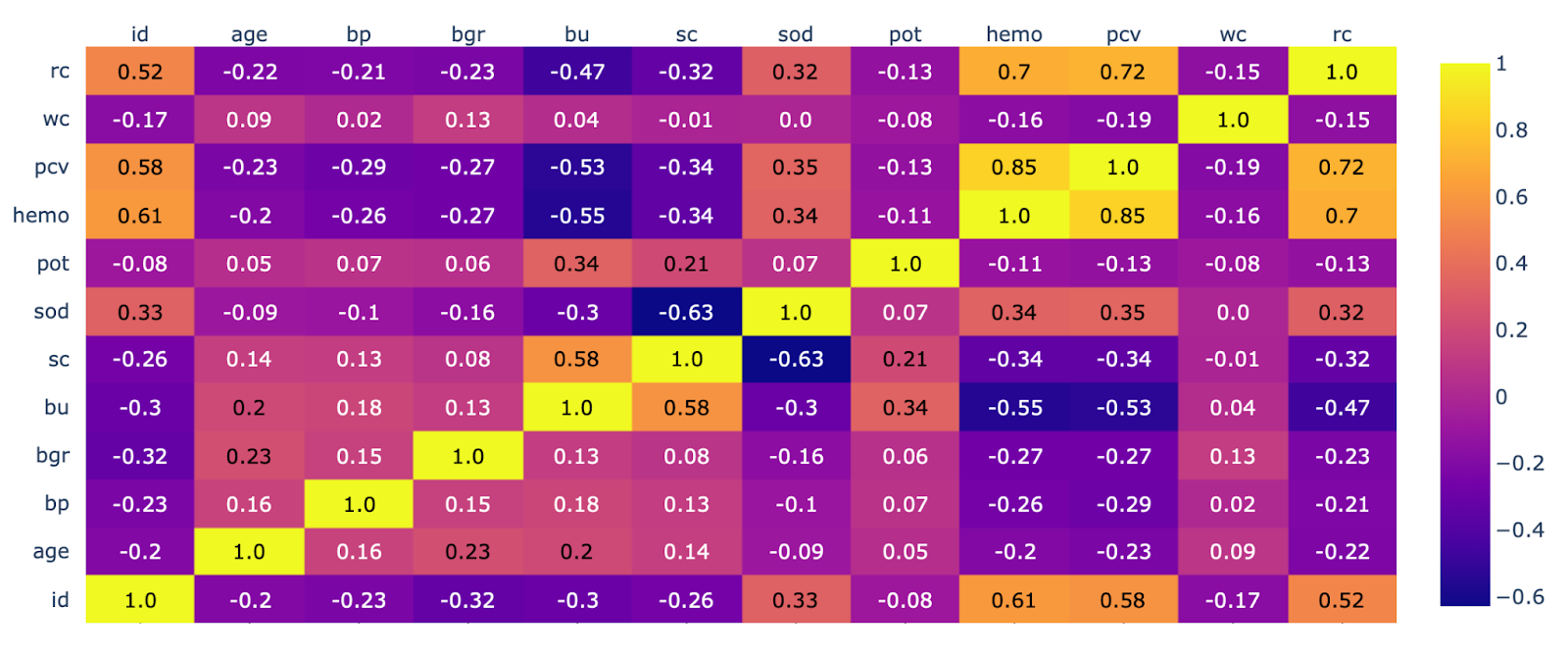
Lastly, we noticed our data has outliers. We have decided to keep outliers because a risk factor we discussed earlier: *Frequent use of medications that can damage the kidneys*. It’s possible our patients have been taking medication that has been altering their biological features. Because of this, we decided to keep the outliers

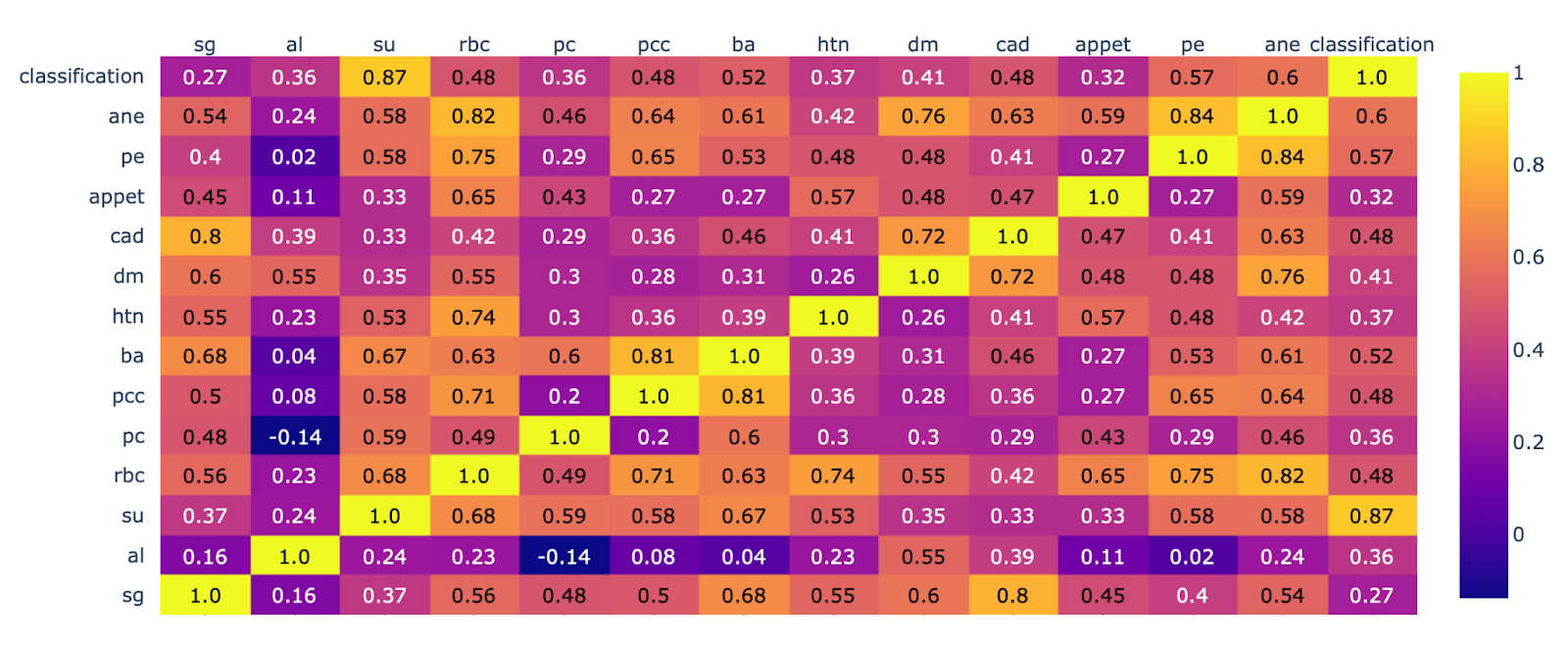
# **Data Visualization**

First thing we do is create a bar chart to see how many patients we have diagnosed with chronic kidney disease. We can see that the majority of the patients in the record are diagnosed with it



Next, we want to observe the correlation between features. We used 2 heatmaps, one observing correlation with numerical data, and one with categorical data

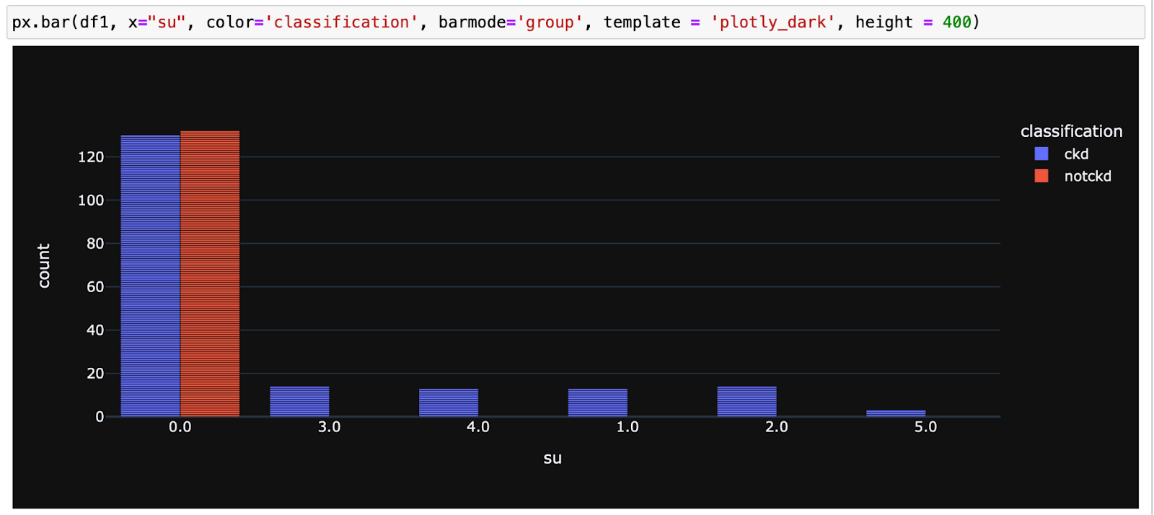


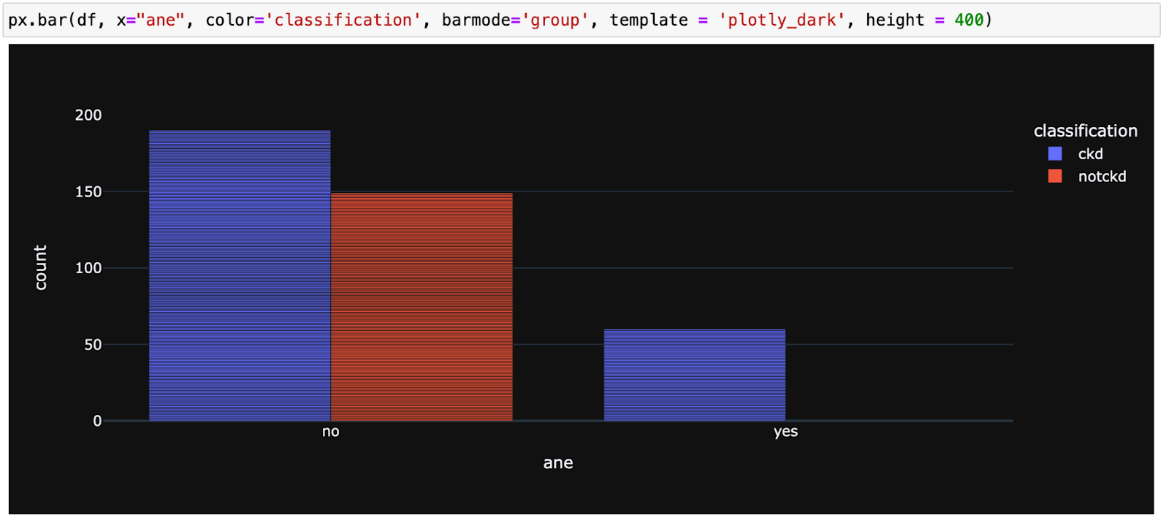


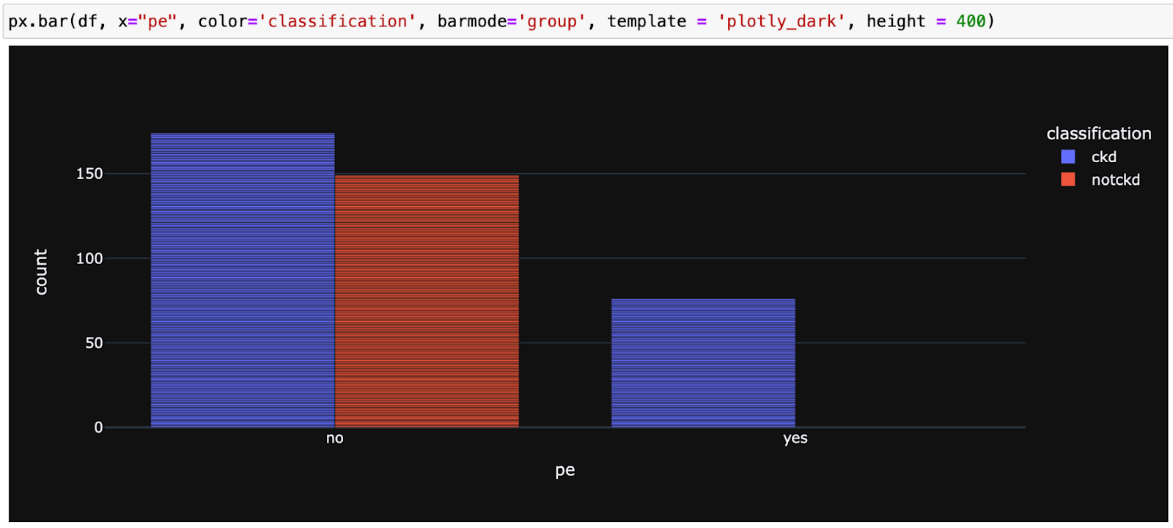
Each square shows the correlation between the variables on each axis. Correlation ranges from -1 to +1. Values closer to zero means there is no linear trend between the two variables. The closer to 1 the correlation is the more positively correlated they are; that is as one increases so does the other and the closer to 1 the stronger this relationship is (like PCV and Hemoglobin). A correlation closer to -1 is similar, but instead of both increasing, one variable will decrease as the other increases (like Albumin and Pus Cell)

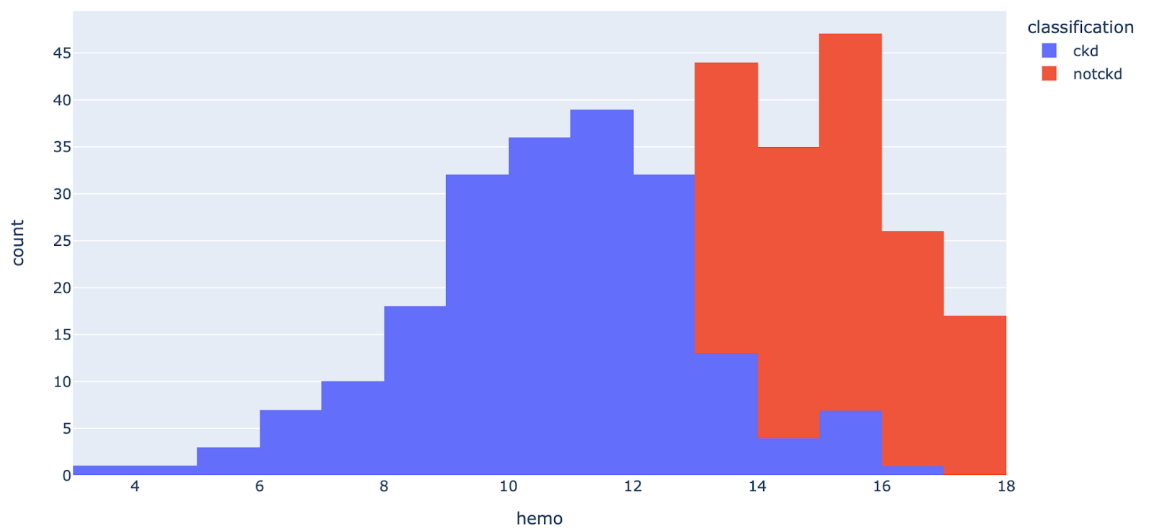
The diagonals are all 1/yellow because those squares are correlating each variable to itself (so it's a perfect correlation). For the rest the larger the number and lighter the color the higher the correlation between the two variables. The plot is also symmetrical about the diagonal since the same two variables are being paired together in those squares.

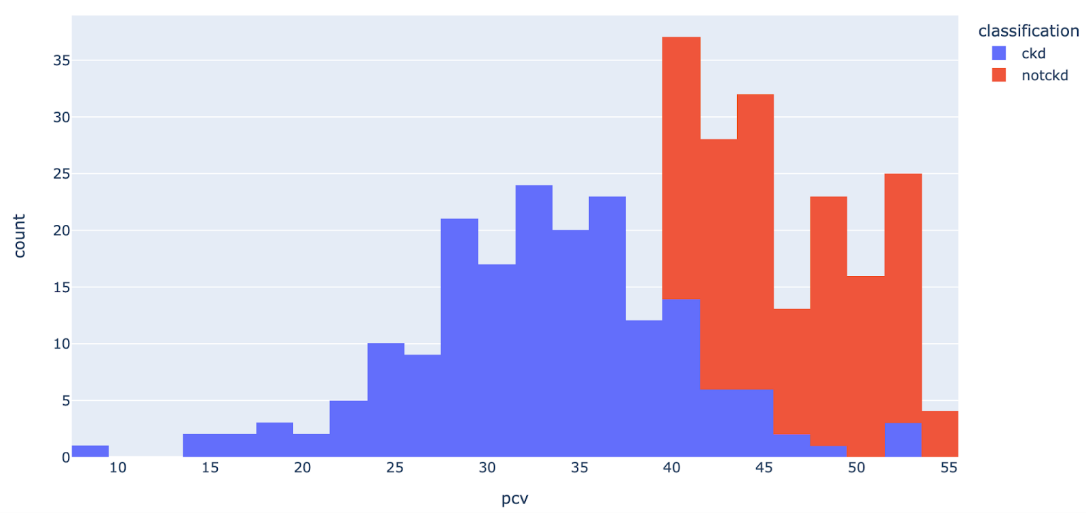
**Plot Charts**

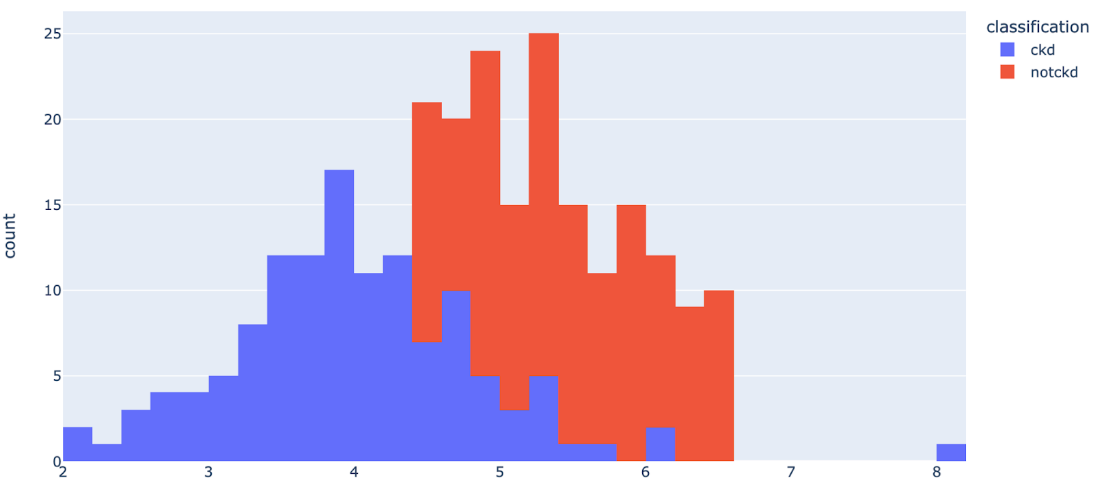


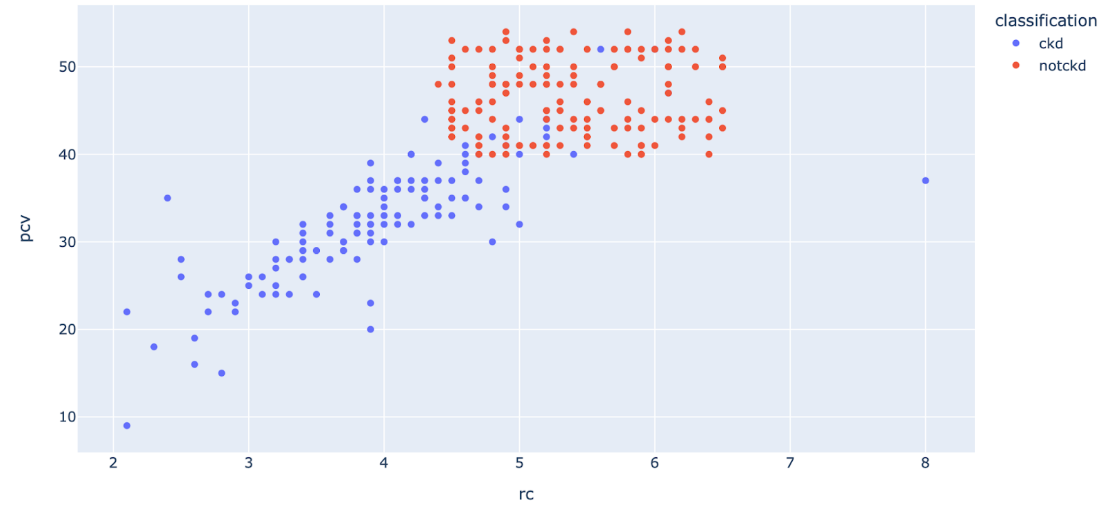




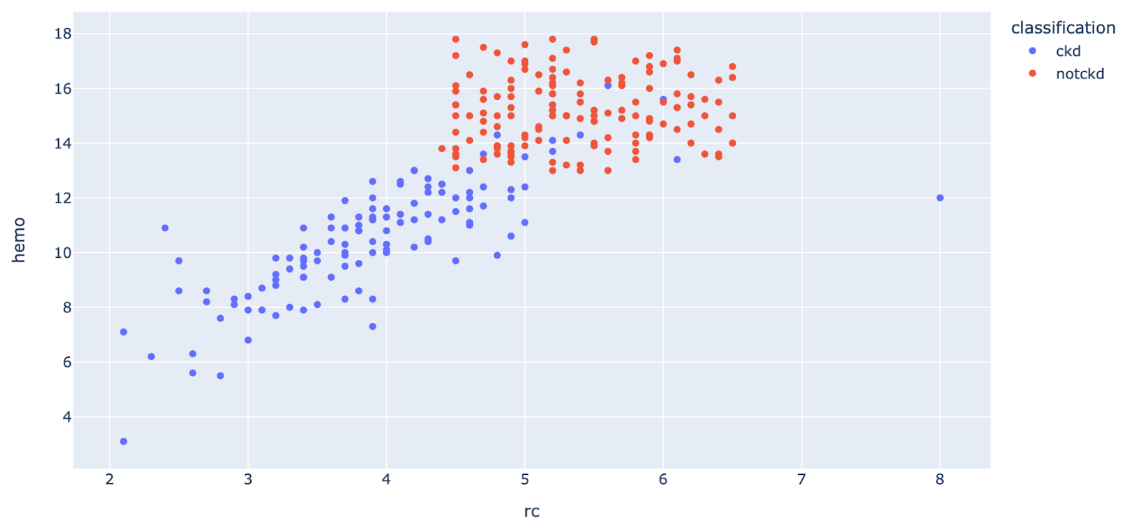






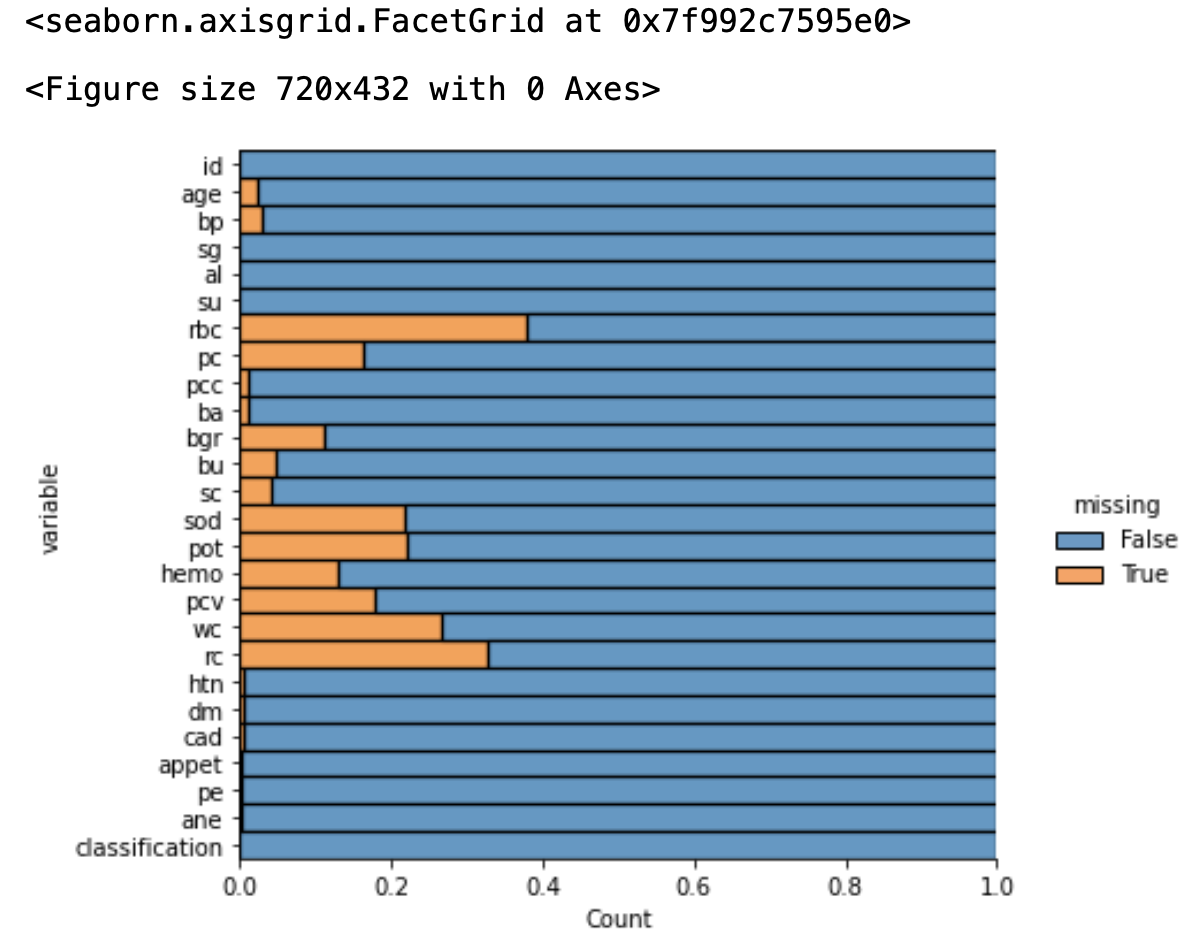
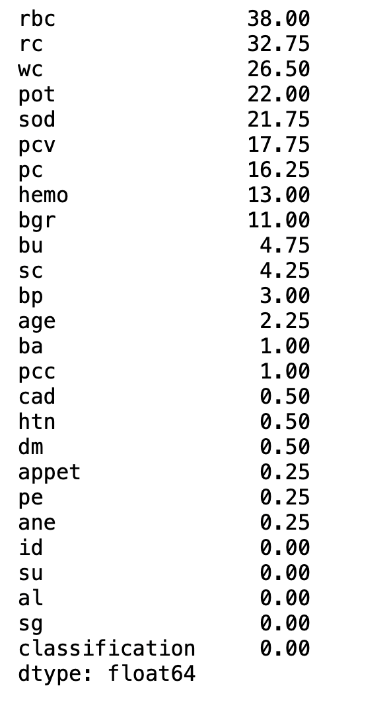






# **Preprocessing**

We want to experiment with the data with and without normalization. Before we can do that, we have to explore the missing values in the dataset. We implemented a method to return the percentage of missing data, and we can see that we have quite a lot



From where we retrieved this dataset, the author recommended we remove every rows with a null value detected, however, this is super destructive, and could leave us with very little data to test and train. Instead, we did the following:

1. Drop the null value for columns that contain less than 5%
2. Fill the numerical values in the columns with the mean
3. Fill the categorical values in the columns with the mode

After doing this, we have **364** records left to experiment with

# **The experiments**

We will now begin to classify our data. We will use 3 models:

* KNN Nearest Neighbor
* Logistic Regression
* Decision Trees (Hyper Parameter Tuning)

We split our 20% of our data into training data, and the rest of the 80% are testing data. We experimented our algorithm 2 times, one without normalization, and one with.

**The Process**

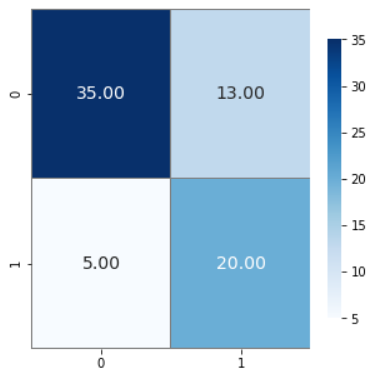
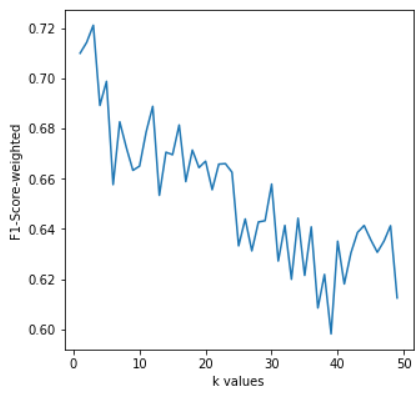
* Create a looping method to find the optimal k
* Test its accuracy and display it on a confusion matrix
* Use logistic regression and tests its training and testing accuracy
* Tuning the hyper parameter with a decision trees and test accuracy

**The Parameters**

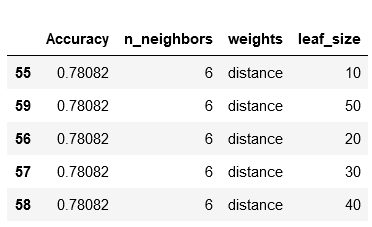
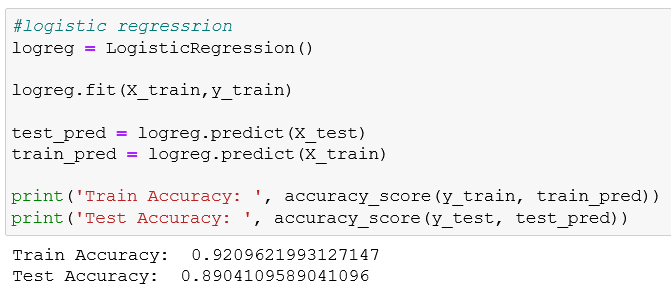
* Random State = 42
* Cross Validation Value = 5
* Tune the hyperparameter by looping method
* Decision Tree Leaf Size = 10, 20, 30, 40, 50
* Find Best K Values = 1 - 50

**Without Normalization**

Our algorithm did not fare well here. First, we found an optimal k = 3, but when testing its accuracy, it only gave us around 75%, using logistic regression improved the testing accuracy to 90%, but the decision tree gave us only 78% accuracy

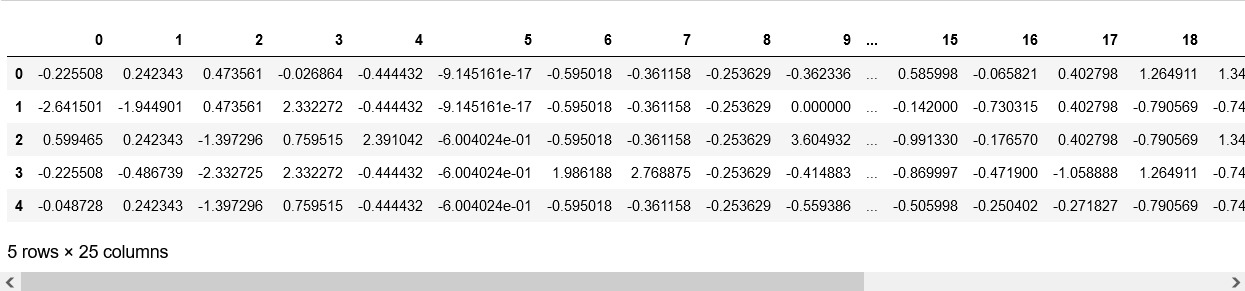


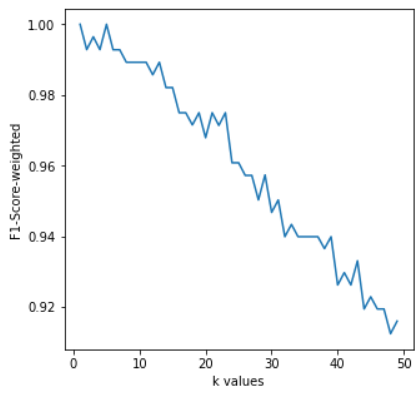
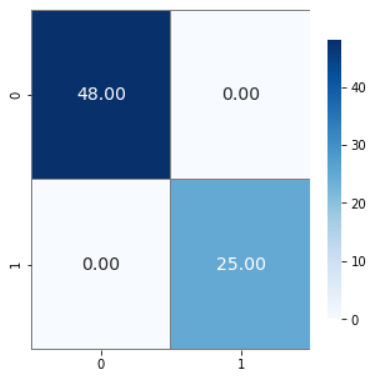
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1 Score | Support |
| Not CKD | 0.61 | 0.60 | 0.69 | 25 |
| CKD | 0.88 | 0.73 | 0.80 | 48 |
| Accuracy |  |  | 0.75 | 73 |
| Macro Avg | 0.74 | 0.76 | 0.74 | 73 |
| Weighted Avg | 0.76 | 0.75 | 0.76 | 73 |



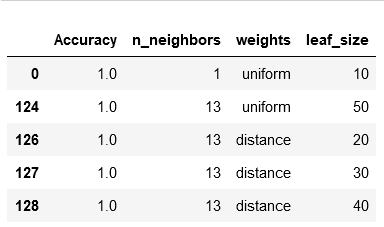
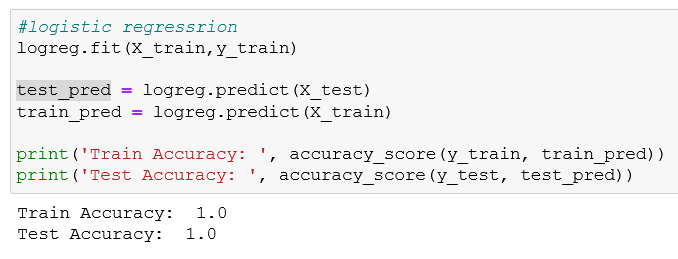
**With Normalization**

We normalized our data with standard scaler, and we can see all the values has been switched to between -3 and 3. After running our algorithms again, we can see that it was able to achieve 100% in all models perfectly! Optimal k = 1



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1 Score | Support |
| Not CKD | 1.0 | 1.0 | 1.0 | 25 |
| CKD | 1.0 | 1.0 | 1.0 | 48 |
| Accuracy |  |  | 1.0 | 73 |
| Macro Avg | 1.0 | 1.0 | 1.0 | 73 |
| Weighted Avg | 1.0 | 1.0 | 1.0 | 73 |



# **Conclusion**

We have successfully studied and understood the biological behavior and domain for chronic kidney disease, and our algorithm were able to predict the diseases in all 3 models with 100% after normalizing it with standard scaler.