**Assignment 3**

**EARLY PREDICTION OF DIABETES RISK**

1. **What is the purpose of loading the dataset at the beginning of your project?**

Ans. Loading the dataset first brings all the patient health information into your analysis environment. This allows you to then explore the data, clean it, and use it to train your machine learning model for diabetes prediction.

1. **What kind of information do you think is stored in this dataset?**

Ans. Based on its name, I expect to find information like Glucose levels, Blood Pressure, and Age. This data could help answer questions about factors contributing to diabetes risk and who is more susceptible.

1. **If the dataset did not load properly, what possible problems could you check?**

Ans. If the data didn't load, I would check if the file path is correct, ensure the file name is spelled exactly right, and verify the file itself isn't corrupted or empty.

1. **What are the names of the columns (features) you saw in the dataset?**

Ans. The columns are Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction, Age, and Outcome. Each represents a health measurement or demographic detail for a patient.

1. **What kind of values do you notice in the first few rows? Are they numbers, text, or something else?**

Ans. In the first few rows, I notice mostly whole numbers and some decimal numbers. The Outcome column contains binary values, either 0 or 1.

1. **Which 2 or 3 columns do you think might be most useful for predicting whether a person has diabetes?**

Ans. Glucose and BMI seem very useful because they are direct indicators of metabolic health. Age might also be important as diabetes risk often increases with age.

1. **Do you notice anything strange, surprising, or possibly wrong in the first few rows of the data?**

Ans. Yes, I noticed some columns like Glucose, BloodPressure, SkinThickness, Insulin, and BMI have zero values. These are surprising and likely incorrect, as these health measurements cannot realistically be zero.

1. **Based on the first few rows, what kind of questions would you like to answer using this data?**

Ans. I'd like to know if high glucose levels alone strongly predict diabetes. Also, I'm curious if older individuals or those with a higher BMI are at significantly greater risk.

1. **How many records (rows) are in the dataset?**

Ans. There are 768 records in the dataset. Each row represents a unique patient and their health information.

1. **How many columns are there, and what does each one seem to represent?**

Ans. There are 9 columns. Pregnancies is number of times pregnant, Glucose is blood sugar, BloodPressure is blood pressure, SkinThickness is triceps skinfold thickness, Insulin is insulin level, BMI is body mass index, DiabetesPedigreeFunction is a genetic score, Age is patient's age, and Outcome is diabetes status.

1. **Did any column have missing data? How do you know?**

Ans. No, according to the "Non-Null Count" which showed 768 for every column, there were no explicitly missing data points (NaN values).

1. **What are the different types of data in the columns? Why does that matter?**

Ans. Columns are either whole numbers (integers like Pregnancies, Glucose) or decimals (floats like BMI, DiabetesPedigreeFunction). Knowing this is useful for choosing appropriate statistical methods and machine learning algorithms.

1. **Based on the structure, do you think this dataset is ready to use for analysis? Why or why not?**

Ans. No, the dataset is not fully ready. While it has no explicit missing values, the presence of unrealistic zero values in several key health columns needs to be fixed before analysis.

1. **Which column has the highest average (mean) value? What does this tell you about that feature?**

Ans. Insulin has the highest average value, which tells me that insulin levels generally have a much larger numerical range compared to other features in this dataset.

1. **Do you notice any columns that have 0 as the minimum value? Is that realistic?**

Ans. Yes, Glucose, BloodPressure, SkinThickness, Insulin, and BMI all have a minimum value of 0. This is not realistic, as these measurements cannot be truly zero in a living person.

1. **Which columns show the most variation (look at the “std” or standard deviation)? Why does variation matter?**

Ans. Insulin and Glucose show high standard deviations. High variation means the values are widely spread out, indicating a diverse range of measurements among patients for these features.

1. **Based on the 25%, 50%, and 75% values (percentiles), which columns seem to have balanced distributions?**

Ans. Pregnancies appears to have a relatively balanced distribution based on its quartiles. The values are not extremely skewed, suggesting a more even spread across its range.

1. **Which features do you think will be most important in predicting the outcome (diabetes or not)? Why?**

Ans. Glucose and BMI will likely be most important. Their minimum values of zero suggest they contain critical information that, once corrected, will directly relate to diabetes status.

1. **Why is it a good idea to make a separate copy of your dataset before making changes?**

Ans. Making a copy is safer because it protects your original data from accidental changes. If something goes wrong during cleaning or transformation, you can always revert to the untouched original.

1. **What do you think could happen if we don’t focus on specific columns for cleaning and just use all the columns together?**

An. If we don't focus on specific columns, we might mistakenly apply cleaning methods to columns that don't need them, or miss problems in columns that do. This could lead to incorrect data and flawed analysis.

1. **Why is it important to check for missing values in your dataset before doing analysis or prediction?**

Ans. Checking for missing values is important because incomplete information can lead to biased or inaccurate machine learning models. A model trained on incomplete data might make unreliable predictions.

1. **Which columns in your dataset had missing values? How many were missing in each?**

Ans. Based on the isnull().sum() output, none of the columns had explicit missing values (NaNs). All columns showed a count of 0 missing values.

1. **Do you think it’s okay for health-related data like “Glucose” or “BloodPressure” to have a value of 0 or be empty? Why or why not?**

Ans. No, it's not okay for Glucose or BloodPressure to be 0 or empty. These are vital signs that cannot be zero in a living person, so a 0 likely represents unrecorded or missing data.

1. **If you ignored missing values completely and trained a machine learning model, what problems might happen?**

Ans. If ignored, the model might learn incorrect patterns, leading to poor predictions. For example, it might incorrectly associate a zero glucose level with a healthy outcome, making it unreliable for diabetes detection.

1. **What can you say about the distribution of the 'Glucose' and 'BloodPressure' values?**

Ans. Both Glucose and BloodPressure histograms show a peak at zero, indicating many unrealistic zero values. Once these are addressed, their distributions appear more centered, but still with some spread.

1. **Which columns seem to have many zero values (e.g., Insulin, SkinThickness)?**

Ans. Insulin and SkinThickness also have many zero values. Zero is not a valid measurement for these, likely representing unrecorded or missing data rather than actual zero levels.

1. **Look at the 'Age' and 'BMI' histograms. What age group or BMI range do most patients fall under?**

Ans. Most patients fall into younger to middle age groups, with a peak around 20-30 years. The BMI distribution appears somewhat bell-shaped, with most patients in the healthy to overweight range.

1. **Check the 'Outcome' column. Are there more diabetic or non-diabetic patients in the dataset?**

Ans. There are more non-diabetic patients (Outcome 0) than diabetic patients (Outcome 1) in the dataset. This imbalance matters because a model might become biased towards the larger group.

1. **What data quality issues do you think should be addressed before using this data in a machine learning model?**

Ans. The primary data quality issue is the presence of unrealistic zero values in several critical health columns. These need to be imputed or handled before any machine learning.

1. **Why is it important to replace zero values in health-related columns like Blood Pressure, Glucose, or BMI?**

Ans. Replacing zero values is crucial because a 0 BMI or 0 Glucose is medically impossible. These zeros are placeholders for missing data, and keeping them would severely distort analysis and model predictions.

1. **What method did you use to replace the zero values, and why is it a good choice?**

Ans. I used the median to replace the zero values. The median is a good choice because it is less affected by extreme values or outliers compared to the mean, providing a more robust imputation.

1. **After cleaning the data, what changes did you observe in the histograms of any two columns?**

Ans. After cleaning, the histograms for Glucose and BMI no longer had large spikes at zero. Their distributions became more realistic, showing a continuous spread of values that better represent actual measurements.

1. **What would happen if you didn’t fix the zero values? How would it affect future analysis or predictions?**

Ans. If zero values weren't fixed, the machine learning model would interpret them as actual measurements, leading to incorrect patterns and highly inaccurate predictions. It would confuse the model about healthy versus unhealthy ranges.

1. **Which column do you think has the most unusual values, and why?**

Ans. Insulin likely has the most unusual values. Its histogram shows a very sharp peak at zero and then a long tail, indicating many missing values and a wide range for the non-zero measurements.

1. **How many rows and columns are present in your dataset? What does each row represent? What does each column represent?**

Ans. My dataset has 768 rows and 9 columns. Each row represents a single patient's medical record, and each column represents a specific health measurement or characteristic for that patient.

1. **Why is it important to check the shape of a dataset before doing any analysis or machine learning?**

Ans. Checking the shape is important to confirm you have enough data for training and to ensure all expected features are present. It helps catch issues like incomplete data loading early on.

1. **If the shape showed fewer columns than expected, what possible issues could have caused this?**

Ans. Fewer columns could be caused by an incorrect file path, a corrupted CSV file where columns were not parsed correctly, or a problem during the data loading process itself.

1. **Imagine you are working with a dataset that has 5,000 rows and only 2 columns. Do you think this is good enough for training a model? Why or why not?**

Ans. 5,000 rows is a good amount of data, but only 2 columns might not be enough features. A model usually needs more diverse information to learn complex patterns for accurate predictions.

1. **If your dataset had 0 rows after loading, what would be your first steps to fix the issue?**

Ans. My first steps would be to double-check the file name and its exact location. Then, I would inspect the data file itself to ensure it's not empty or corrupted.

1. **How many people in the dataset have diabetes and how many do not?**

Ans. In the dataset, 500 people do not have diabetes, and 268 people do have diabetes.

1. **Which group is larger in the dataset – people with diabetes or without diabetes?**

Ans. The group of people without diabetes is significantly larger in the dataset compared to the group with diabetes.

1. **Why do you think it's important to check the number of diabetic and non-diabetic cases before building a model?**

Ans. It's important to check the class balance because if one group is much smaller, the model might not learn enough about it, leading to poor predictions for that group.

1. **If one group is much bigger than the other, what kind of problem could that cause in prediction?**

Ans. If one group is much bigger, the model might become biased towards predicting the majority class. This means it could be very good at predicting "no diabetes" but poor at identifying actual diabetes cases.

1. **How would you describe this dataset to someone who wants to build a diabetes prediction app?**

Ans. This dataset contains health metrics for 768 patients, with a notable imbalance where non-diabetic cases are almost twice as many as diabetic cases.

1. **Which two features (columns) in the scatter matrix seem to have the strongest positive relationship?**

Ans. Glucose and Insulin appear to have a strong positive relationship, as points generally increase together. BMI and SkinThickness also show a positive correlation.

1. **Do you notice any features that do not seem related to others?**

Ans. Pregnancies seems to have less clear relationships with many other features. Its scatter plots often show scattered points without a strong trend.

1. **Were there any unusual points (outliers) visible in the scatter matrix?**

Ans. Yes, some features like Insulin and BloodPressure show unusual points, or outliers, appearing far from the main cluster of data. These could be rare but valid measurements or data entry errors.

1. **What do the histograms on the diagonal line of the scatter matrix tell you about the data?**

Ans. The histograms on the diagonal show the distribution of each individual feature. For example, the Age histogram indicates that most patients are in younger to middle age ranges.

1. **Based on the scatter matrix, which features do you think might be most useful for predicting diabetes?**

Ans. Glucose and BMI seem most useful. Their scatter plots with Outcome often show some separation between the two groups, suggesting they are good discriminators.

1. **Which features show a clear difference between people with and without diabetes?**

Ans. Glucose and BMI show a clear difference. People with diabetes tend to have higher glucose levels and higher BMI values compared to those without diabetes.

1. **Is there any feature that does not help much in separating diabetic and non-diabetic individuals?**

Ans. SkinThickness and BloodPressure appear to have less clear separation between diabetic and non-diabetic individuals. Both groups show similar patterns across their ranges.

1. **What extra insight do you gain by using hue='Outcome' in the pairplot that you do not get in the plain pairplot?**

Ans. Using hue='Outcome' allows you to visually see how well features separate the two classes. For example, with Glucose, the color-coded plot clearly shows higher concentrations of diabetic patients at higher glucose values.

1. **If you had to choose only one pairplot to include in your final project report, which one would you choose and why?**

Ans. I would choose the pairplot with hue='Outcome'. It's more informative because it directly shows the relationship between features and the target variable, which is crucial for a prediction project.

1. **How can the pairplot with hue='Outcome' help you decide which features might be important for predicting diabetes?**

An. It helps by showing visual separation. If the colors (representing diabetic/non-diabetic) are clearly clustered in different areas for a feature, like Glucose, that feature is likely important for prediction.

1. **Which two features in the dataset show the strongest positive relationship?**

Ans. BMI and SkinThickness show the strongest positive relationship, meaning as one increases, the other tends to increase.

1. **Which feature shows the strongest relationship with the Outcome (diabetes)?**

Ans. Glucose shows the strongest positive relationship with the Outcome. This is important because higher glucose levels are a direct indicator of diabetes.

1. **Were there any pairs of features that showed little or no relationship (value close to 0)?**

Ans. Yes, several pairs showed little to no relationship, with correlation values close to 0. For example, Pregnancies and Insulin had a very weak connection.

1. **What insights did the heatmap give you that you could not easily see in the pair plot?**

Ans. The heatmap provided precise numerical correlation coefficients, allowing for a quick, quantitative comparison of relationships across all pairs that was harder to eyeball from the scatter plots alone.

1. **How can this heatmap help in choosing the right features for building a prediction model?**

Ans. The heatmap helps by clearly showing which features are most strongly correlated with the Outcome. Based on this, Glucose and BMI would be top choices for predicting diabetes.

1. **Why did we need to scale the data before building a machine learning model?**

Ans. We needed to scale the data because features like Glucose and Insulin have very different value ranges. Scaling ensures that features with larger values don't disproportionately influence the model's distance calculations.

1. **Which columns were selected for scaling, and which column was left out? Why?**

Ans. All columns except 'Outcome' were selected for scaling. 'Outcome' was left out because it's the target variable, a binary class label, not a feature that needs numerical scaling.

1. **What do you think would happen if we skipped the scaling step? How might it affect the model’s performance?**

Ans. If scaling was skipped, features with larger numerical ranges would dominate the distance calculations in KNN. This would lead to a biased model that performs poorly because it prioritizes some features unfairly.

1. **After scaling, do the actual values of the features (like BMI or Glucose) still have the same meaning? Explain.**

Ans. After scaling, the numerical values change, but the underlying meaning and relationships between the data points remain the same. Scaling only transforms the range, not the inherent information.

1. **How is the new table after scaling different from the original one? What’s one benefit of creating a separate table instead of replacing the original?**

Ans. The new table has standardized values, centered around zero. A benefit of creating a separate table is data safety, allowing you to easily revert to the original unscaled data if needed.

1. **Why do you think it is important to balance the number of diabetic and non-diabetic cases in the dataset before training a model?**

Ans. It's important to balance the classes because an imbalanced dataset can cause the model to be biased towards the majority class, leading to poor performance on the minority class.

1. **What problems could arise if we did not apply SMOTE and trained the model on unbalanced data? Explain with a simple example.**

Ans. Without SMOTE, the model might predict "no diabetes" most of the time because it's the most frequent outcome. This means it would miss many actual diabetes cases, leading to a high number of false negatives.

1. **After applying SMOTE, what changes did you observe in the number of samples for each class? Write the numbers before and after.**

Ans. Before SMOTE, the training data had 400 non-diabetic and 214 diabetic cases. After SMOTE, both classes were balanced with 400 samples each.

1. **Do you think synthetic (artificial) data created by SMOTE is as useful as real data? Why or why not?**

Ans. Synthetic data is useful for balancing, but not as good as real data. It helps the model learn patterns for the minority class, but it might not capture the full complexity or variability of real-world data.

1. **How does balancing the data help improve the fairness or accuracy of your final machine learning model?**

Ans. Balancing the data improves fairness and accuracy by giving the model enough examples of the minority class. This prevents it from simply predicting the majority class and helps it correctly identify actual diabetes cases.

1. **Why do we test the KNN model with different values of k (number of neighbors)?**

Ans. We test with different 'k' values to find the optimal number of neighbors that yields the best model performance. Changing 'k' helps us understand the trade-off between model simplicity and complexity.

1. **Based on your results, which value of k gave the best performance on the test data? Was this value also the best for training data? Why might they be different?**

Ans. The best 'k' for testing was 14. This was not the best for training data; training accuracy is often highest at smaller 'k' values because the model memorizes the training data more closely.

1. **Did you notice any k values where the model performed very well on training data but poorly on testing data? What could this tell us about the model?**

Ans. Yes, at smaller 'k' values (e.g., k=1), training accuracy was very high while testing accuracy was low. This tells us the model was overfitting, meaning it memorized the training data too well but couldn't generalize.

1. **Why is it important to compare both training accuracy and testing accuracy when building a machine learning model? What could go wrong if we only looked at training accuracy?**

Ans. Comparing both is crucial to assess generalization. If we only looked at training accuracy, we might pick an overfitting model that performs poorly on new, unseen data in the real world.

1. **Imagine you are recommending a KNN model for predicting diabetes in a real hospital setting. Which value of k would you choose and why? Justify your answer with your testing results.**

Ans. I would choose k=14 because it gave the highest testing accuracy. This value indicates the model generalizes best to unseen patient data, which is critical for reliable predictions in a hospital setting.

1. **Which value(s) of k gave the highest training accuracy?**

Ans. The highest training accuracy was 1.0 (100%) which occurred at k=1. This is important to observe as it indicates the model perfectly memorized the training data at this k value.

1. **Which value(s) of k gave the highest testing accuracy?**

Ans. The highest testing accuracy was 0.2153, which occurred at k=14. Testing accuracy is crucial because it shows how well the model performs on new, unseen data.

1. **Did the best training score and best testing score happen at the same value of k?**

Ans. No, the best training score (k=1) and best testing score (k=14) did not happen at the same k. This difference indicates that the model behaves differently when memorizing versus generalizing.

1. **Is there a big gap between training and testing accuracy?**

Ans. Yes, there is a significant gap between training and testing accuracy, especially at smaller k values. This means the model is overfitting, memorizing the training data instead of learning generalizable patterns.

1. **Which k value did you finally choose for your model? Why?**

Ans. I chose k=14 for the final model. Although training accuracy was lower, this k value yielded the highest testing accuracy, indicating the best generalization performance for predicting new cases.

1. **What do you observe about the training accuracy as the value of K increases?**

Ans. As K increases, the training accuracy generally decreases. This tells me that the model's ability to memorize the training data lessens as it considers more neighbors.

1. **At which value of K does the testing accuracy seem to be the highest?**

Ans. The testing accuracy seems highest around K=14. This might be a good choice for the final model because high testing accuracy means it predicts diabetes correctly for new patients.

1. **Is there a big difference between training and testing accuracy at some K values?**

Ans. Yes, there is a big difference, especially at small K values. This indicates the model is overfitting, meaning it performs well on seen data but poorly on unseen data.

1. **Why is it important to compare both training and testing accuracy when selecting a machine learning model?**

Ans. It's important to compare both to avoid overfitting. If we only look at training accuracy, we might choose a model that performs well on old data but fails on new, real-world data.

1. **What value of K (number of neighbors) did you choose for the KNN model, and why?**

Ans. I chose K=14 for the KNN model because it yielded the highest testing accuracy, which is crucial for good generalization. The training accuracy was lower but more realistic at this value.

1. **What is the final accuracy of your model on the test data?**

Ans. The final accuracy of my model on the test data is 0.1845. This accuracy is quite low and suggests the model might not be high enough for making reliable real-world predictions.

1. **Is your model performing better on training data or testing data? What does this tell you about the model?**

Ans. My model is performing significantly better on training data than on testing data. This tells me the model is overfitting, meaning it has memorized the training data too well but struggles with new data.

1. **How did applying SMOTE (oversampling) affect your model’s performance?**

Ans. Applying SMOTE balanced the training data, which helps the model learn from the minority class. This likely improved its ability to detect diabetes cases, as it prevented bias towards the non-diabetic majority.

1. **If you had more time or resources, what would you try next to improve the model’s accuracy or usefulness?**

Ans. I would try tuning the KNN model's settings more rigorously, exploring other classification algorithms like Random Forest, and potentially incorporating more relevant features if available.

1. **What does your model’s precision score tell you?**

Ans. My model's precision score of 1.0 tells me that when it predicted a person has diabetes, it was always correct. It did not raise any false alarms.

1. **What does the recall score reveal about your model’s performance?**

Ans. The recall score of 0.0185 reveals that my model was only able to catch a very small fraction of the actual diabetes cases. It missed almost all the true diabetes cases.

1. **Why is the F1 score useful in this project?**

Ans. The F1 score is useful because it provides a balanced measure between precision and recall. It helps understand the overall effectiveness when both correctly identifying cases and avoiding false alarms are important.

1. **How did SMOTE affect the balance of your training data, and do you think it improved your model's performance?**

Ans. SMOTE balanced the training data by creating synthetic samples for the minority class. While it helped the model learn about the minority class, the very low recall suggests it didn't fully improve performance in catching actual cases.

1. **Based on your evaluation (precision, recall, F1), what would you suggest to improve this model’s performance further?**

Ans. To improve, I would suggest trying different classification models that might handle the data better. Also, further hyperparameter tuning and potentially more advanced feature engineering could help improve recall significantly.