Planned analysis Example for Storyboard:

Note: This will need to be summarized and needs more info

The accuracy score is simply the percentage of predictions that are correct. In this case, the model's accuracy score was 0.712, meaning that the model was correct 71.2% of the time.

Logistic Regression is popular due to its accuracy and versatility, which is why it was a great choice for this ML project.

We have already seen one way of validating such a model's performance: its accuracy score. An accuracy score is not always an appropriate or a meaningful performance metric, however.

The following is another way to validate the performance of our classification model.

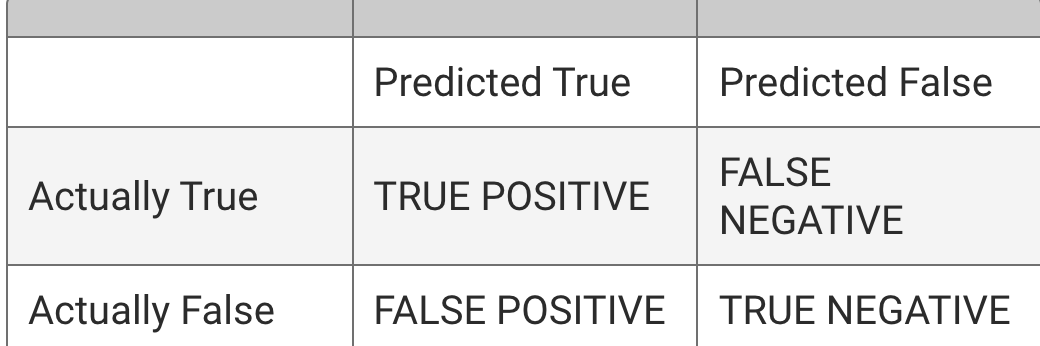
Any given prediction falls under one of two categories: true or false. In the context of fraud detection, a true prediction would mean that the model categorizes the transaction as fraudulent. A false prediction means that the model categorizes the transaction as not fraudulent.

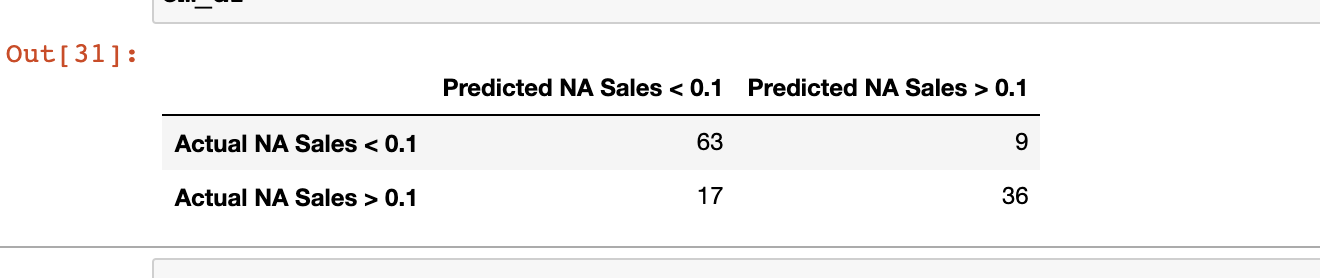
If a transaction is predicted to be fraudulent and is really a fraudulent transaction, it is a true positive (TP).

If a transaction is predicted to be fraudulent but is not fraudulent, it is a false positive (FP). It falsely categorized an innocent transaction as fraudulent.

Similarly, if a transaction is predicted to be non-fraudulent but is actually fraudulent, it is a false negative (FN).

And when a transaction is predicted to be non-fraudulent and is in reality non-fraudulent, it is a true negative (TN).





**Precision,** also known as positive predictive value (PPV), is a measure of this. Precision is obtained by dividing the number of true positives (TP) by the number of all positives (i.e., the sum of true positives and false positives, or TP + FP). Precision = TP/(TP + FP)

To summarize, in machine learning, precision is a measure of how reliable a positive classification is. The following formulation may help you in remembering precision: "I know that the test for cancer came back positive. How likely is it that I have cancer?"

Another way to assess a model's performance is with sensitivity, also called recall. While the term **recall** is more commonly used in machine learning, the two terms are synonymous and will be used interchangeably from this point.

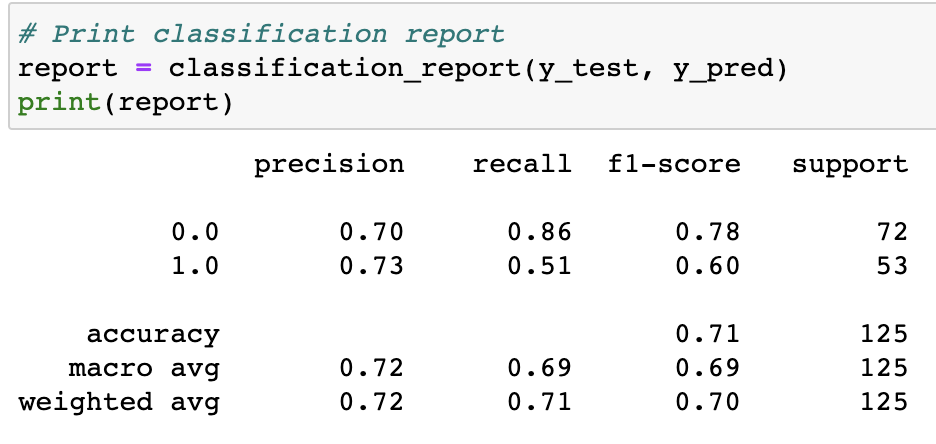
The following formulation may help you understand sensitivity: "I know that I have cancer. How likely is it that the test will diagnose it?" Here is the formula for sensitivity:

Sensitivity = TP/(TP + FN)

In this context, all who have cancer means true positives (those who have cancer and were correctly diagnosed) and false negatives (those who have cancer and were incorrectly diagnosed as not having cancer). **Sensitivity** is a measure of how many people who actually have cancer were correctly diagnosed. Note: **Sensitivity** and **Recall** are used interchangeably.  Sensitivity is more important. A test with high sensitivity means few false negatives, though there may be a high number of false positives.

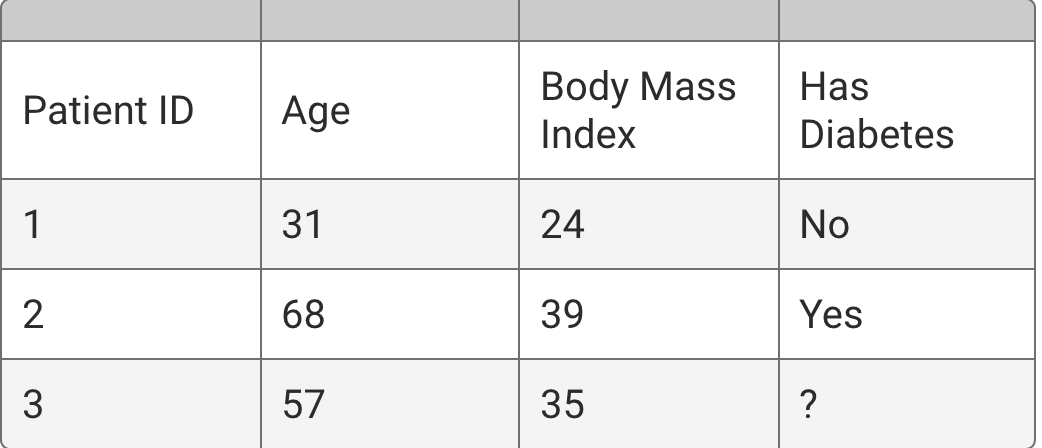
The F1 score, also called the harmonic mean, can be characterized as a single summary statistic of precision and sensitivity. The formula for the F1 score is the following:

2(Precision \* Sensitivity)/(Precision + Sensitivity)



From module:

Supervised learning deals with labeled data. An example of supervised learning might be to predict, based on data from previous patients, whether a new patient has diabetes.



Recall that **regression** is used to predict continuous variables. For example, let's say that we're interested in predicting a person's weight based on factors like height, dietary preferences, and exercise patterns. To accomplish this task, we would collect data on a number of people. The regression model's algorithms would attempt to learn patterns that exist among these factors. If presented with the data of a new person, the model would make a prediction of that person's weight, based on previously learned patterns from the dataset.

**Classification**, on the other hand, is used to predict discrete outcomes. For example. Let's say that we are interested in using a person's traits, such as age, sex, income, and geographic location, to predict how she or he will vote on a particular issue. The outcome, in this case, is whether the person will vote "Yes" or "No." The classification model's algorithms would attempt to learn patterns from the data, and if the model is successful, gain the ability to make accurate predictions for new voters.

In our regression example, the target variable, or what we're trying to predict, is weight. Weight is a continuous variable—a person's weight can be any numerical value within a certain range.

**Features** are the variables used to make a prediction. **Target** is the predicted outcome.

20.2.4

The first model you will use is logistic regression, a popular classification model. She explains that despite its name, logistic regression is actually not a regression model. It is a classification model. With logistic regression, it is possible to try to answer questions such as whether a credit card holder is likely to miss a payment in the next month.

**Logistic regression** predicts binary outcomes, meaning that there are only two possible outcomes. An example of logistic regression might be to decide, based on personal information, whether to approve a credit card application. Multiple variables, such as an applicant's age and income, are assessed to arrive at one of two answers: to approve or to deny the application.

In other words, a logistic regression model analyzes the available data, and when presented with a new sample, mathematically determines its probability of belonging to a class. If the probability is above a certain cutoff point, the sample is assigned to that class. If the probability is less than the cutoff point, the sample is assigned to the other class.

**README DRAFT**

**## Communication Protocols:**

The team’s communication protocols will be relatively the same for all segments of the project. The team is aware of the importance of communicating with each other in order to achieve a successful project outcome. We will continue to respect lines communication, address any disagreements early on, and collectively work on assignments distributed according to function and expertise.

For segment two, we added the following protocols:

* Discuss the team’s goals and objectives during the given segment
* Roles and requirements are clearly defined
* Set regular communication goals, types, and schedules
* Increase group meetings via Zoom to clearly define scope, resources, and timeline
* Continue group messages in Slack
* Milestone status is regularly reported to the team
* Attend office hours to ask questions and ensure the project is on the right track
* Motivate each other
* Recognize and praise a member’s great work

**## Outline of Project:**

**Note: We need to fill the blanks. This is the module’s Rubric**

**Presentation:**

**GitHub Repository Management:**

Note: For better management and organization of the repository, the team agreed to create feature branches of project versus individual branches for team members. As a result, we submit (add) any documents to designated feature branches of the project, and provide detailed description on “commit” message.

**README File:**

**Machine Learning Model:**

The team members are expected to submit the code for the machine learning model, as well as the following:

1. Description of preliminary data preprocessing
2. Description of preliminary feature engineering and preliminary feature selection, including the decision-making process
3. Description of how data was split into training and testing sets
4. Explanation of model choice, including limitations and benefits

**Database Integration:**

1. The team members are expected to present a fully integrated database, including the following:
2. Database stores static data for use during the project
3. Database interfaces with the project in some format (e.g., scraping updates the database)
4. Includes at least two tables (or collections, if using MongoDB)
5. Includes at least one join using the database language (not including any joins in Pandas)
6. Includes at least one connection string (using SQLAlchemy or PyMongo)

**Dashboard:**

A blueprint for the dashboard is created and includes all of the following:

1. Storyboard on a Google Slide(s)
2. Description of the tool(s) that will be used to create the final dashboard
3. Description of interactive elements(s)