ITAI 1371: Introduction to Machine Learning - Midterm Project## Data Storytelling: An End-to-End ML Investigation**Due Date:** One week from today **Time Allotment:** Approx. 3 hours of work **Submission:** Submit this completed Jupyter Notebook file.---### Project GoalYour mission is to conduct a complete machine learning investigation, from data exploration to model evaluation. More than just writing code, you will be a **data storyteller**. Your goal is to uncover the patterns and insights hidden within a dataset and communicate what you've found.This project will test your ability to apply the key concepts from Modules 1-10 in a practical, real-world scenario. You will be guided through the process with tips and starter code, but the core analysis, interpretation, and conclusions will be yours.### Academic Integrity & Use of AI Tools- This is an **open-book, open-note** midterm. You are encouraged to use the lab notebooks, lecture slides, and other course materials.- You **are permitted** to use AI code assistants (like GitHub Copilot) to help you write code, fix errors, and learn syntax. This is a valuable real-world skill.- You **are NOT permitted** to

use AI to generate entire sections of analysis, interpretation, or answers to reflective questions. The goal is for *you* to demonstrate understanding.**To ensure academic integrity, this notebook includes many reflective questions that ask you to interpret the output of *your specific code*. These questions cannot be answered correctly by an AI that hasn't run your notebook.**---### Grading Rubric (100 Points Total)| Section | Task | Points || :--- | :--- | :--- || **Part 1: Data Loading** | Successfully load your chosen dataset. | 5 || **Part 2: EDA** | Create and interpret at least two relevant visualizations. | 20 || | Write a clear interpretation of the story your plots tell. | || **Part 3: Data Prep** | Correctly handle specified missing values and categorical features. | 15 || | Justify your choices in the markdown cells. | || **Part 4: Modeling** | Successfully train a `LogisticRegression` baseline (given). | 15 || | Successfully train a second model of your choice. | || **Part 5: Evaluation** | Calculate and compare accuracy for both models. | 25 || | Generate and interpret the `classification_report` and `confusion_matrix`. | || | Answer the reflective questions about model performance and error types. |

|| **Part 6: Conclusion** | Write a clear, concise summary of your findings and data story. | 15 || | Address the key questions in the conclusion prompt. | || **Overall** | Code is clean, commented, and runs without errors. All markdown cells are filled out. | 5 |

Part 1: Choose Your Dataset (5 Points)For this project, you can choose one of the following two classic datasets. Both are classification problems.1. **Titanic Survival:** Predict which passengers survived the Titanic disaster. (You are familiar with this from our lab).2. **Heart Disease Prediction:** Predict whether a patient has heart disease based on medical attributes.**Instructions:**1. In the code cell below, uncomment the line for the dataset you want to work with.2. Run the cell to load the data into a pandas DataFrame called `df`.3. Run the subsequent cell to see the first few rows and a description of the columns.

```
import pandas as pd
import numpy as np
# --- CHOOSE YOUR DATASET ---
# Uncomment one of the two lines below to select your dataset
# Option 1: Titanic Dataset
dataset_url = 'https://raw.githubusercontent.com/datasciencedojo/datasets/maste
# Option 2: Heart Disease Dataset
# dataset_url = 'https://raw.githubusercontent.com/plotly/datasets/master/heart

# --- LOAD THE DATA ---
# This code will load the dataset from the URL you selected above
try:
    df = pd.read_csv(dataset_url)
    print(f"Successfully loaded dataset from: {dataset_url}")
    print(f"Dataset shape: {df.shape}")
```

```
except Exception as e:
    print(f"Error loading dataset: {e}")
    print("Please make sure you have selected a valid URL.")
```

```
Successfully loaded dataset from: https://raw.githubusercontent.com/datasciencedo
Dataset shape: (891, 12)
```

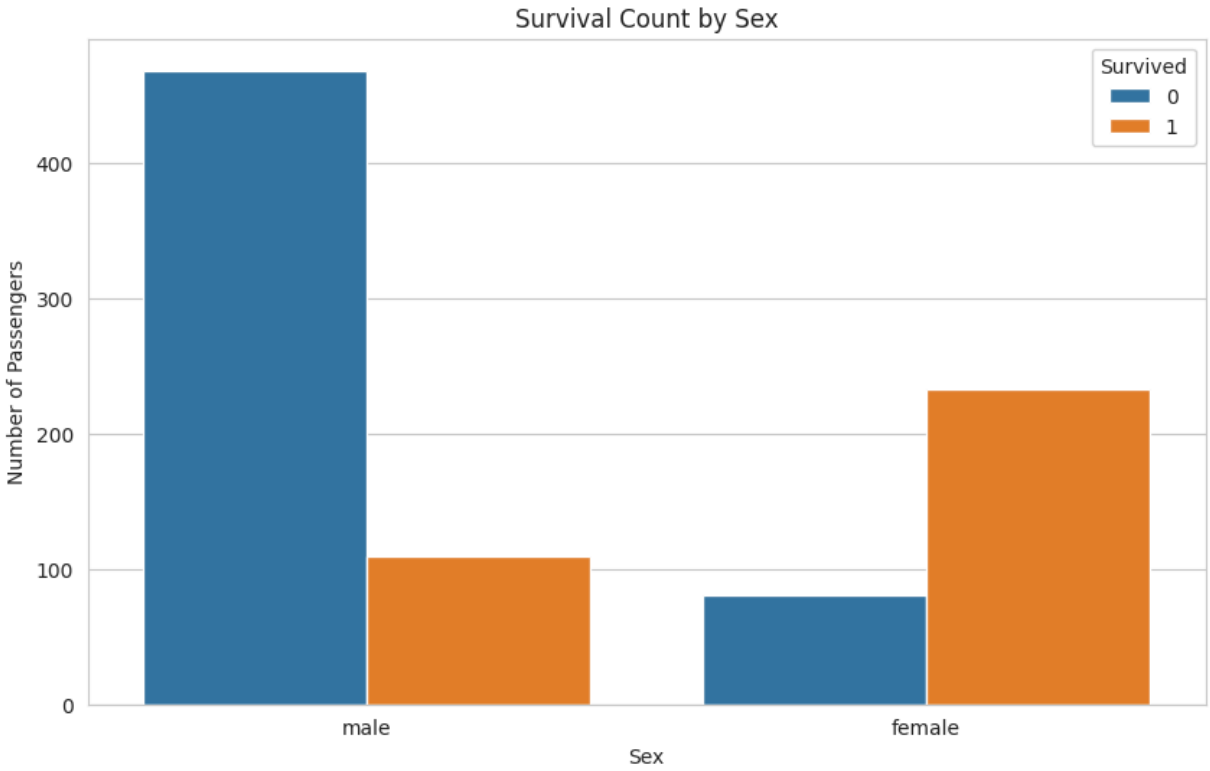
Data OverviewRun the cell below to display the first 5 rows of your dataset, a list of its columns, and a brief description of what each column means.

```
# Display the first 5 rows of the dataframeprint("--- First 5 Rows ---")print(d
```

Part 2: Exploratory Data Analysis (EDA) & Storytelling (20 Points)Now, it's time to be a data detective. Before you can model the data, you must understand it. What secrets does it hold?**Your Task:**1. **Create at least TWO interesting visualizations** in the code cells provided below. You can create more if you like. * Use libraries like `matplotlib` or `seaborn`. * Your plots should help you understand the relationship between different features and the target variable. * **Tip:** Think about the questions we asked during the Titanic lab (e.g., "How does survival rate differ by gender?" or "What is the age distribution of survivors?"). Ask similar questions of your dataset.2. **Interpret your visualizations** in the markdown cell provided. Explain what you see and what story your plots are starting to tell.

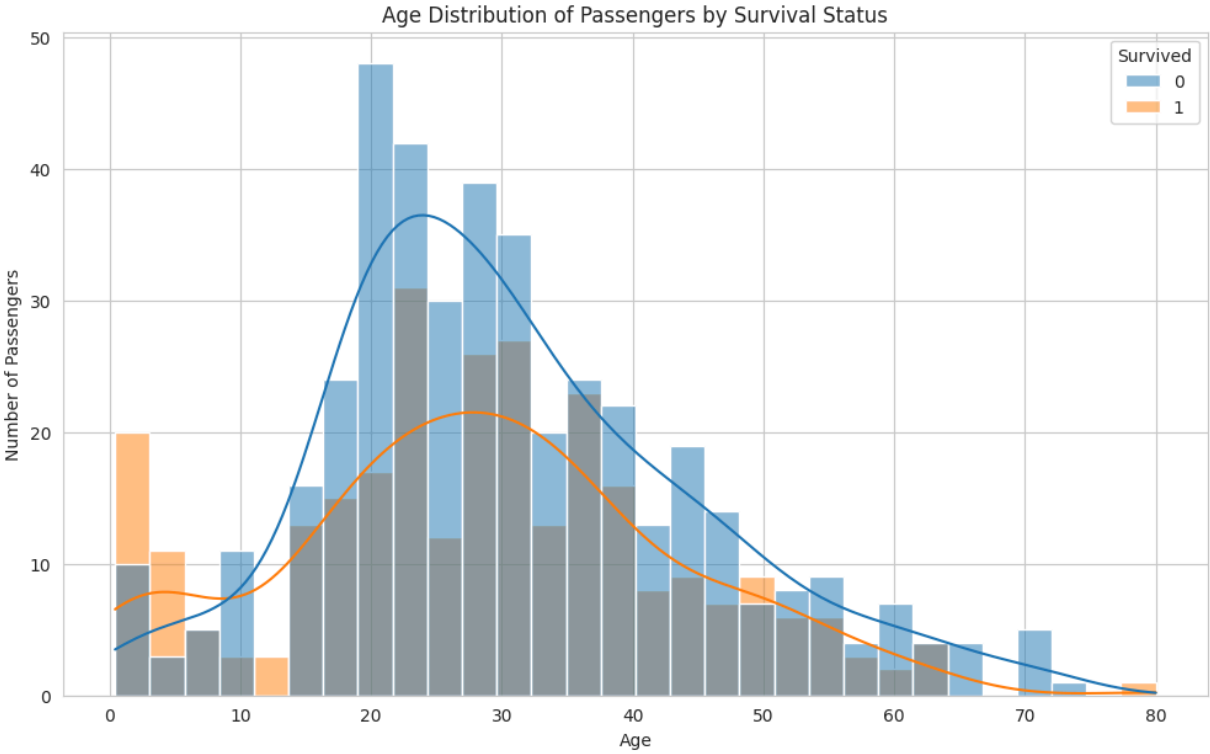
Visualization 1**Instructions:** Create your first plot in the cell below. Make sure to give it a title and label your axes! Good plots are easy to read.

```
import matplotlib.pyplot as plt
import seaborn as sns
sns.set_style('whitegrid')
# --- ENTER YOUR CODE HERE ---
# Create your first visualization. Some ideas:
# - A countplot to see the distribution of the target variable.
# - A barplot to compare a feature against the target (e.g., 'sex' vs 'survivec
# - A histogram or KDE plot to see the distribution of a numerical feature (e.g
plt.figure(figsize=(10, 6))
sns.countplot(x='Sex', hue='Survived', data=df)
plt.title('Survival Count by Sex')
plt.xlabel('Sex')
plt.ylabel('Number of Passengers')
# --- END OF YOUR CODE ---
plt.show()
```



Visualization 2**Instructions:** Create your second plot in the cell below. Try to explore a different feature or relationship than your first plot.

```
# --- ENTER YOUR CODE HERE ---
# Create your second visualization. Some ideas:
# - A boxplot to see the distribution of a numerical feature across different c
# - A heatmap of correlations between numerical features.
# - A facet grid to explore relationships across multiple categories.
plt.figure(figsize=(12, 7))
sns.histplot(data=df, x='Age', hue='Survived', kde=True, bins=30)
plt.title('Age Distribution of Passengers by Survival Status')
plt.xlabel('Age')
plt.ylabel('Number of Passengers')
# --- END OF YOUR CODE ---
plt.show()
```



Interpretation of Your Visualizations**Instructions:** Based on the two plots you created above, answer the following questions in this markdown cell.1. **What did you plot?** (Briefly describe your two visualizations).2. **What story do your plots tell?** (What initial insights or patterns did you discover? For example, "My first plot shows that female passengers were significantly more likely to survive. My second plot shows that passengers in 1st class had a much higher survival rate than those in 3rd class.")3. **What is one hypothesis you can form based on your EDA?** (e.g., "I hypothesize that age and passenger

class will be the most important features for predicting survival.")**---ENTER YOUR ANSWERS BELOW ---**1. ...2. ...3. ...

1. The first graph is a bar chart that shows how many survived for both genders. The second is a barchar and line graph combo.
2. The first graph shows more women survived than men. The second graph shows how many passangers survived along the lines of age as its saparated by every 10 years and the line chart shows a smoothed out bell curve of the bar chart.
3. The most important feature for survival as shown in both graphs is that women had the highest chance of survival. the second graph gives us more nuance in that age also a key factor in survival. More young children and young adults survived when compared to the elderly.

Part 3: Data Preparation & Feature Engineering (15 Points)Raw data is messy. We need to clean it up before feeding it to our models.**Your Task:**1. **Handle Missing Values:** I've provided code to check for missing values. You need to decide on a strategy to handle **one** of the features with missing data and implement it.2. **Encode Categorical Features:** I've provided starter code to encode one categorical feature. You need to encode **one more** categorical feature of your choice.3. **Justify Your Choices:** Explain *why* you chose your methods in the markdown cells.

```
# Check for missing valuesprint("--- Missing Values Before ---")print(df.isnull
```

Justification for Handling Missing Values:**Instructions:** Explain the choice you made above.1. **Which feature did you choose?**2. **What method did you use to handle the missing values (e.g., fill with median, mode, or drop)?**3. ***Why was this an appropriate method for this feature?****--- ENTER YOUR ANSWERS BELOW ---**1. ...2. ...3. ...

1. Age.
2. Median imputation was used to replace the missing values in the age column.

3. The second graph shows a clustering from 20 to 40, so using the median for age makes sense as it accounts for the elderly and children unlike averaging with the mean would create noise as would mode.

```
# --- Starter Code for Encoding ---# For the Titanic dataset, we encode 'Sex'.
```

Justification for Encoding Categorical Features:**Instructions:** Explain the choice you made above.1. **Which feature did you choose to encode?**2. **What encoding method did you use (e.g., map, pd.get_dummies)?**3. ***Why was this the right method? If you used `get_dummies`, why is `drop_first=True` often a good idea?****--- ENTER YOUR ANSWERS BELOW ---**1. ...2. ...3. ...

1. Embarked.
2. pd.get_dummies. It kept those whoe embarked and marked as "C" as a standard and created two new columns for "Q" and "S".
3. Theres only 3 catagories and from what I understand theres no heirarchy between them so encoding them here makes sense.

Part 4 & 5: Modeling and Evaluation (40 Points)Now for the main event! Let's train some models and see how well they can predict outcomes.**Your Task:**1. **Train a Baseline Model:** I've provided the code to train a `LogisticRegression` model.2. **Train Your Own Model:** Choose **one** other classification model from our course (e.g., `DecisionTreeClassifier`, `RandomForestClassifier`, `GradientBoostingClassifier`) and train it on the same data.3. **Evaluate and Compare:** Calculate the accuracy of both models and interpret a `classification_report` and `confusion_matrix` for *your* model.4. **Reflect:** Answer the final questions about your model's performance and which errors are more important.

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report, confusion_ma

# --- Define Features (X) and Target (y) ---
# This is a sample feature set. You may need to adjust it based on the columns
# Make sure all columns are numeric and have no missing values.
# Drop non-numeric or irrelevant columns before defining features
df_model = df.copy()
df_model = df_model.select_dtypes(include=np.number).dropna()

if 'PassengerId' in df_model.columns:
    df_model = df_model.drop(columns=['PassengerId'])

# Define target variable name based on dataset
target_col = 'Survived' if 'Survived' in df_model.columns else 'target'
X = df_model.drop(target_col, axis=1)
y = df_model[target_col]

# --- Split Data ---
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random

print(f"Training set has {X_train.shape[0]} samples.")
print(f"Test set has {X_test.shape[0]} samples.")
print(f"Features: {X.columns.tolist()}")
```

```
Training set has 571 samples.
Test set has 143 samples.
Features: ['Pclass', 'Age', 'SibSp', 'Parch', 'Fare']
```

Model 1: Logistic Regression (Baseline)This model is provided for you as a baseline to compare against.

```
# Train the baseline modellog_reg = LogisticRegression(max_iter=1000, random_st
```

Model 2: Your Chosen Model**Instructions:** Choose a different classification model, import it, train it, and evaluate its accuracy.

```
# --- ENTER YOUR CODE HERE ---# 1. Import your chosen model class# from sklearr
```

Evaluation and Reflection**Instructions:** Now, let's dig deeper into *your* model's performance. Generate a `classification_report` and

`confusion_matrix` for the model you just trained. Then, answer the reflective questions.

```
    # --- ENTER YOUR CODE HERE ---# Generate and print the classification report fc
```

Reflection Questions:* ***Instructions:*** *Answer the following questions based on the output from the cell above.1.* ***Which model performed better, the baseline or yours?*** *Was it a big difference?2.* ***Look at the `classification_report` for your model. What are the precision and recall for the positive class (1)?*** *(Just state the values).3.* ***Interpret the precision and recall. In the context of your chosen dataset, what do these numbers mean?*** *(e.g., "A recall of 0.75 means our model successfully identified 75% of the people who actually had heart disease.")4.* ***Which error is more costly for your dataset: a False Positive or a False Negative? Explain your reasoning.*** *(There is no single right answer, it depends on your justification). * **Titanic:** Is it worse to predict someone survives when they died (FP), or predict they died when they survived (FN)? * **Heart Disease:** Is it worse to tell a healthy person they have heart disease (FP), or tell a sick person they are healthy (FN)?**--- ENTER YOUR ANSWERS BELOW ---**1. ...2. ...3. ...4. ...

1. My random forest model performed better than the linear regression model. The difference may not have been big but it is evident.
2. My model performed at .82 percent or 3.7% over the baseline of .79 percent.
3. Precision for survival for class 1 was .78 percent and this tells me that my random forest model predicts "Survival" its correct 78% of the time when compared to the baseline of .72 percent. Recall is also higher in my model at .74 percent compared to the baseline of .70 percent. My model corectly identifies .74 percent of survivors were actually predicted as such.
4. In my opinion a false negative is a much worse error than a false positive . I would rather a model that give me a lower predictive survival rate than a slightly higher actual survival rate. If I were to base a critical decision based off of these two choices, id rather spend the man power, resources and material to launch a search and rescue and have a chance at saving someone who could be saved.

Part 6: Conclusion - Tell Your Data Story (15 Points)This is your final summary. Bring together everything you've learned

from your investigation.**Instructions:**Write 2-3 paragraphs summarizing your project. Your summary should be a narrative that tells the story of your data. Address the following points:- **What was the main goal of your project?** - **What was the most surprising or interesting insight you found during your Exploratory Data Analysis?**- **Which features seemed to be the most important for making predictions?**- **How well did your best model perform, and what are its limitations?** (Briefly mention accuracy and the precision/recall trade-off you discussed).- **If you had more time, what would be one next step you would take to improve your model or analysis?****---- ENTER YOUR CONCLUSION BELOW ---**...

The main goal of this exercise was to find the trends in the data from the titanic, but to also find nuances from the data set. Something that surprised me was that there was a framework to think about how to fill in data. Before I would have thought that the data is sacrosanct. If its missing data then it can not under any circumstances be altered for any reason. Diving deeper into the EDA section it cleared up some of the reasons why it was ok to use things like median, mean and modes of things to fill in some data.

The Titanic data set revealed to me how powerful one feature, in this case gender, was to such a determininistic data set. Other factors like class and age had a role to play sure but a blanket 70% chance of survival for women was a stark revealation. My model worked well but theres room for improvement. The missing data from embarked, and other details not recorded at the time kind of puts a limit on my models efficiacy. That being said what insights I can gain from it could spark conversations that could lead to future improvements.