



# Student Performance Predictor

A complete end-to-end Data Science project in Google Colab.

In [2]: `from google.colab import files`  
`uploaded = files.upload()`

Choose Files

No file chosen

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

Saving card\_transdata.csv to card\_transdata.csv

In [3]: `import pandas as pd`  
  
`# Load the CSV file`  
`df = pd.read_csv('card_transdata.csv')`  
`df.head()`

Out[3]:

	distance_from_home	distance_from_last_transaction	ratio_to_median_purchase_price	repe
0	57.877857	0.311140	1.945940	
1	10.829943	0.175592	1.294219	
2	5.091079	0.805153	0.427715	
3	2.247564	5.600044	0.362663	
4	44.190936	0.566486	2.222767	

In [4]: `# Data Exploration`  
`df.info()`  
`df.describe()`  
`df.shape`  
`df.columns`

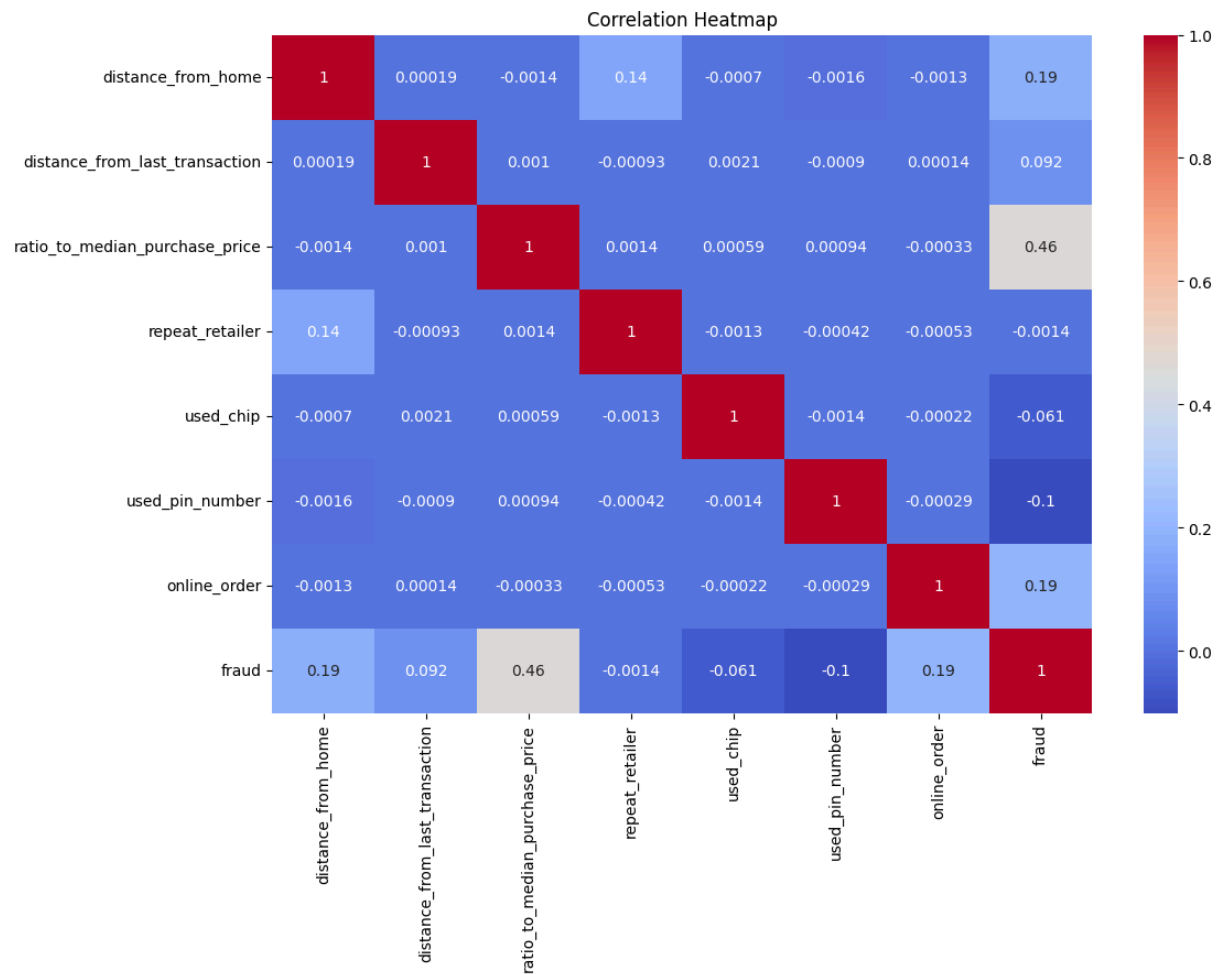
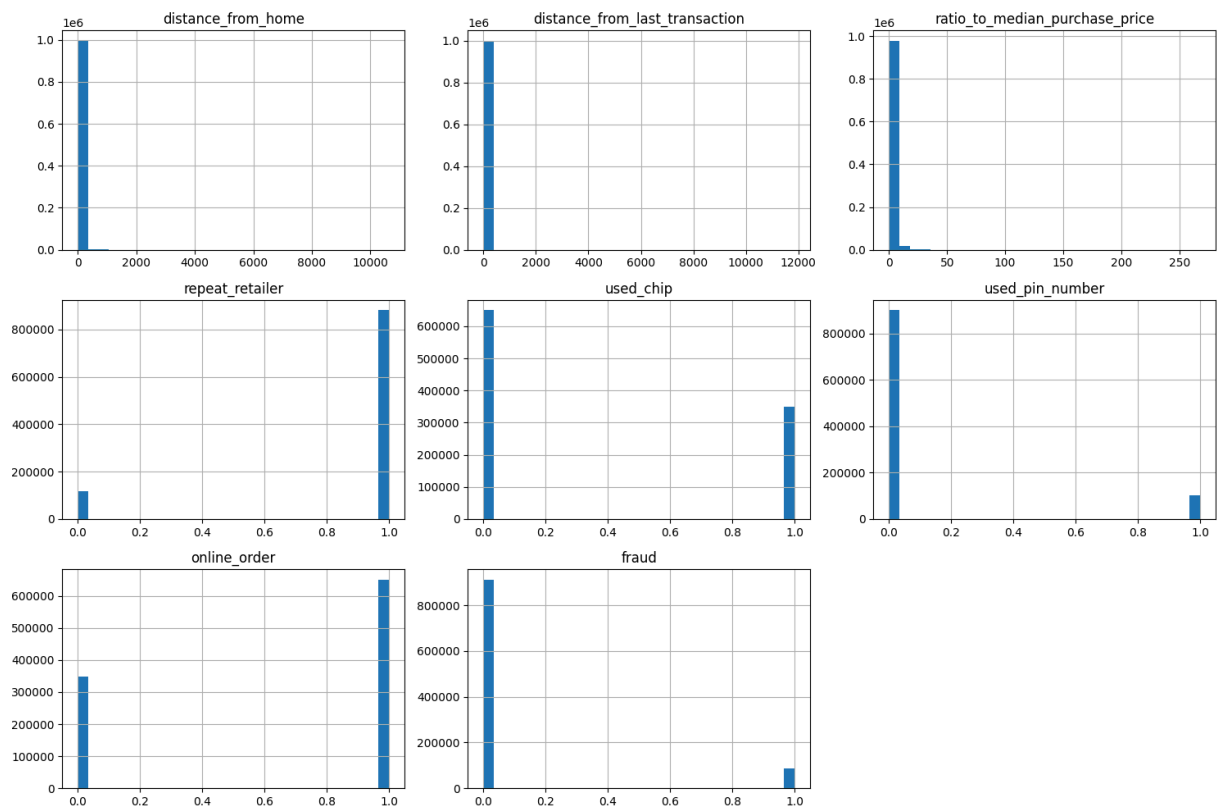
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000000 entries, 0 to 999999
Data columns (total 8 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   distance_from_home                    1000000 non-null float64
1   distance_from_last_transaction        1000000 non-null float64
2   ratio_to_median_purchase_price        1000000 non-null float64
3   repeat_retailer                       1000000 non-null float64
4   used_chip                             1000000 non-null float64
5   used_pin_number                       1000000 non-null float64
6   online_order                          1000000 non-null float64
7   fraud                                 1000000 non-null float64
dtypes: float64(8)
memory usage: 61.0 MB
```

```
Out[4]: Index(['distance_from_home', 'distance_from_last_transaction',  
              'ratio_to_median_purchase_price', 'repeat_retailer', 'used_chip',  
              'used_pin_number', 'online_order', 'fraud'],  
             dtype='object')
```

```
In [5]: # Check for Missing Values and Duplicates  
print(df.isnull().sum())  
print(f'Duplicates: {df.duplicated().sum()}')
```

```
distance_from_home          0  
distance_from_last_transaction  0  
ratio_to_median_purchase_price  0  
repeat_retailer             0  
used_chip                   0  
used_pin_number             0  
online_order                0  
fraud                       0  
dtype: int64  
Duplicates: 0
```

```
In [6]: import matplotlib.pyplot as plt  
import seaborn as sns  
  
# Histogram  
df.hist(bins=30, figsize=(15, 10))  
plt.tight_layout()  
plt.show()  
  
# Correlation Heatmap  
plt.figure(figsize=(12,8))  
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')  
plt.title("Correlation Heatmap")  
plt.show()
```



```
In [7]: # Identify Target and Features
        target = 'your_target_column_name' # Change this to the actual target
```

```
features = [col for col in df.columns if col != target]
```

```
In [8]: # Convert Categorical Columns to Numerical
cat_cols = df.select_dtypes(include=['object']).columns
df[cat_cols] = df[cat_cols].apply(lambda x: x.astype('category').cat.codes)
```

```
In [9]: # One-Hot Encoding
df = pd.get_dummies(df, drop_first=True)
```

```
In [14]: # Step 1: Set your actual target column
target = 'fraud' # Replace 'fraud' with your actual target column name

# Step 2: Perform feature scaling
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
X_scaled = scaler.fit_transform(df.drop(columns=target)) # X_scaled is your scaled

# Optionally, set up the Labels (y)
y = df[target]
```

```
In [15]: # Train-Test Split
from sklearn.model_selection import train_test_split

X = X_scaled
y = df[target]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_sta
```

```
In [19]: # Model Building
from sklearn.ensemble import RandomForestClassifier

model = RandomForestClassifier()
model.fit(X_train, y_train)
```

```
Out[19]: ▼ RandomForestClassifier ⓘ ?
RandomForestClassifier()
```

```
In [20]: # Evaluation
from sklearn.metrics import classification_report, accuracy_score

y_pred = model.predict(X_test)
print("Accuracy:", accuracy_score(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

Accuracy: 0.99999

	precision	recall	f1-score	support
0.0	1.00	1.00	1.00	182557
1.0	1.00	1.00	1.00	17443
accuracy			1.00	200000
macro avg	1.00	1.00	1.00	200000
weighted avg	1.00	1.00	1.00	200000

```
In [21]: # Make Predictions from New Input
sample = [0.5] * X.shape[1] # Replace with actual feature values
prediction = model.predict([sample])
print("Prediction:", prediction)
```

Prediction: [0.]

```
In [22]: # Convert to DataFrame and Encode
new_data = pd.DataFrame([sample], columns=df.drop(columns=target).columns)
```

```
In [23]: # Predict the Final Grade (or Output)
predicted_value = model.predict(new_data)
print("Predicted:", predicted_value[0])
```

Predicted: 0.0

```
/usr/local/lib/python3.11/dist-packages/sklearn/utils/validation.py:2732: UserWarning: X has feature names, but RandomForestClassifier was fitted without feature names
warnings.warn(
```

```
In [24]: # Install Gradio
!pip install gradio
```

```

Collecting gradio
  Downloading gradio-5.29.1-py3-none-any.whl.metadata (16 kB)
Collecting aiofiles<25.0,>=22.0 (from gradio)
  Downloading aiofiles-24.1.0-py3-none-any.whl.metadata (10 kB)
Requirement already satisfied: anyio<5.0,>=3.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (4.9.0)
Collecting fastapi<1.0,>=0.115.2 (from gradio)
  Downloading fastapi-0.115.12-py3-none-any.whl.metadata (27 kB)
Collecting ffmpeg (from gradio)
  Downloading ffmpeg-0.5.0-py3-none-any.whl.metadata (3.0 kB)
Collecting gradio-client==1.10.1 (from gradio)
  Downloading gradio_client-1.10.1-py3-none-any.whl.metadata (7.1 kB)
Collecting groovy~=0.1 (from gradio)
  Downloading groovy-0.1.2-py3-none-any.whl.metadata (6.1 kB)
Requirement already satisfied: httpx>=0.24.1 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.28.1)
Requirement already satisfied: huggingface-hub>=0.28.1 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.31.1)
Requirement already satisfied: jinja2<4.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (3.1.6)
Requirement already satisfied: markupsafe<4.0,>=2.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (3.0.2)
Requirement already satisfied: numpy<3.0,>=1.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (2.0.2)
Requirement already satisfied: orjson~=3.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (3.10.18)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-packages (from gradio) (24.2)
Requirement already satisfied: pandas<3.0,>=1.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (2.2.2)
Requirement already satisfied: pillow<12.0,>=8.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (11.2.1)
Requirement already satisfied: pydantic<2.12,>=2.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (2.11.4)
Collecting pydub (from gradio)
  Downloading pydub-0.25.1-py2.py3-none-any.whl.metadata (1.4 kB)
Collecting python-multipart>=0.0.18 (from gradio)
  Downloading python_multipart-0.0.20-py3-none-any.whl.metadata (1.8 kB)
Requirement already satisfied: pyyaml<7.0,>=5.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (6.0.2)
Collecting ruff>=0.9.3 (from gradio)
  Downloading ruff-0.11.9-py3-none-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (25 kB)
Collecting safehttpx<0.2.0,>=0.1.6 (from gradio)
  Downloading safehttpx-0.1.6-py3-none-any.whl.metadata (4.2 kB)
Collecting semantic-version~=2.0 (from gradio)
  Downloading semantic_version-2.10.0-py2.py3-none-any.whl.metadata (9.7 kB)
Collecting starlette<1.0,>=0.40.0 (from gradio)
  Downloading starlette-0.46.2-py3-none-any.whl.metadata (6.2 kB)
Collecting tomlkit<0.14.0,>=0.12.0 (from gradio)
  Downloading tomlkit-0.13.2-py3-none-any.whl.metadata (2.7 kB)
Requirement already satisfied: typer<1.0,>=0.12 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.15.3)
Requirement already satisfied: typing-extensions~=4.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (4.13.2)
Collecting uvicorn>=0.14.0 (from gradio)

```

Downloading uvicorn-0.34.2-py3-none-any.whl.metadata (6.5 kB)  
 Requirement already satisfied: fsspec in /usr/local/lib/python3.11/dist-packages (from gradio-client==1.10.1->gradio) (2025.3.2)  
 Requirement already satisfied: websockets<16.0,>=10.0 in /usr/local/lib/python3.11/dist-packages (from gradio-client==1.10.1->gradio) (15.0.1)  
 Requirement already satisfied: idna>=2.8 in /usr/local/lib/python3.11/dist-packages (from anyio<5.0,>=3.0->gradio) (3.10)  
 Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.11/dist-packages (from anyio<5.0,>=3.0->gradio) (1.3.1)  
 Requirement already satisfied: certifi in /usr/local/lib/python3.11/dist-packages (from httpx>=0.24.1->gradio) (2025.4.26)  
 Requirement already satisfied: httpcore==1.\* in /usr/local/lib/python3.11/dist-packages (from httpx>=0.24.1->gradio) (1.0.9)  
 Requirement already satisfied: h11>=0.16 in /usr/local/lib/python3.11/dist-packages (from httpcore==1.\*->httpx>=0.24.1->gradio) (0.16.0)  
 Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.28.1->gradio) (3.18.0)  
 Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.28.1->gradio) (2.32.3)  
 Requirement already satisfied: tqdm>=4.42.1 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.28.1->gradio) (4.67.1)  
 Requirement already satisfied: hf-xet<2.0.0,>=1.1.0 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.28.1->gradio) (1.1.0)  
 Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas<3.0,>=1.0->gradio) (2.9.0.post0)  
 Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas<3.0,>=1.0->gradio) (2025.2)  
 Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas<3.0,>=1.0->gradio) (2025.2)  
 Requirement already satisfied: annotated-types>=0.6.0 in /usr/local/lib/python3.11/dist-packages (from pydantic<2.12,>=2.0->gradio) (0.7.0)  
 Requirement already satisfied: pydantic-core==2.33.2 in /usr/local/lib/python3.11/dist-packages (from pydantic<2.12,>=2.0->gradio) (2.33.2)  
 Requirement already satisfied: typing-inspection>=0.4.0 in /usr/local/lib/python3.11/dist-packages (from pydantic<2.12,>=2.0->gradio) (0.4.0)  
 Requirement already satisfied: click>=8.0.0 in /usr/local/lib/python3.11/dist-packages (from typer<1.0,>=0.12->gradio) (8.1.8)  
 Requirement already satisfied: shellingham>=1.3.0 in /usr/local/lib/python3.11/dist-packages (from typer<1.0,>=0.12->gradio) (1.5.4)  
 Requirement already satisfied: rich>=10.11.0 in /usr/local/lib/python3.11/dist-packages (from typer<1.0,>=0.12->gradio) (13.9.4)  
 Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas<3.0,>=1.0->gradio) (1.17.0)  
 Requirement already satisfied: markdown-it-py>=2.2.0 in /usr/local/lib/python3.11/dist-packages (from rich>=10.11.0->typer<1.0,>=0.12->gradio) (3.0.0)  
 Requirement already satisfied: pygments<3.0.0,>=2.13.0 in /usr/local/lib/python3.11/dist-packages (from rich>=10.11.0->typer<1.0,>=0.12->gradio) (2.19.1)  
 Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests->huggingface-hub>=0.28.1->gradio) (3.4.2)  
 Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests->huggingface-hub>=0.28.1->gradio) (2.4.0)  
 Requirement already satisfied: mdurl~0.1 in /usr/local/lib/python3.11/dist-packages (from markdown-it-py>=2.2.0->rich>=10.11.0->typer<1.0,>=0.12->gradio) (0.1.2)  
 Downloading gradio-5.29.1-py3-none-any.whl (54.1 MB)  
 54.1/54.1 MB 9.4 MB/s eta 0:00:00  
 Downloading gradio\_client-1.10.1-py3-none-any.whl (323 kB)

```

323.1/323.1 kB 17.9 MB/s eta 0:00:00
Downloading aiofiles-24.1.0-py3-none-any.whl (15 kB)
Downloading fastapi-0.115.12-py3-none-any.whl (95 kB)
95.2/95.2 kB 6.0 MB/s eta 0:00:00
Downloading groovy-0.1.2-py3-none-any.whl (14 kB)
Downloading python_multipart-0.0.20-py3-none-any.whl (24 kB)
Downloading ruff-0.11.9-py3-none-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (11.5 MB)
11.5/11.5 MB 31.8 MB/s eta 0:00:00
Downloading safehttpx-0.1.6-py3-none-any.whl (8.7 kB)
Downloading semantic_version-2.10.0-py2.py3-none-any.whl (15 kB)
Downloading starlette-0.46.2-py3-none-any.whl (72 kB)
72.0/72.0 kB 4.4 MB/s eta 0:00:00
Downloading tomlkit-0.13.2-py3-none-any.whl (37 kB)
Downloading uvicorn-0.34.2-py3-none-any.whl (62 kB)
62.5/62.5 kB 4.3 MB/s eta 0:00:00
Downloading ffmpeg-0.5.0-py3-none-any.whl (6.0 kB)
Downloading pydub-0.25.1-py2.py3-none-any.whl (32 kB)
Installing collected packages: pydub, uvicorn, tomlkit, semantic-version, ruff, python-multipart, groovy, ffmpeg, aiofiles, starlette, safehttpx, gradio-client, fastapi, gradio
Successfully installed aiofiles-24.1.0 fastapi-0.115.12 ffmpeg-0.5.0 gradio-5.29.1 gradio-client-1.10.1 groovy-0.1.2 pydub-0.25.1 python-multipart-0.0.20 ruff-0.11.9 safehttpx-0.1.6 semantic-version-2.10.0 starlette-0.46.2 tomlkit-0.13.2 uvicorn-0.34.2

```

```

In [25]: # Create a Prediction Function
def predict_student_performance(*inputs):
    import numpy as np
    input_array = np.array(inputs).reshape(1, -1)
    input_scaled = scaler.transform(input_array)
    prediction = model.predict(input_scaled)
    return prediction[0]

In [26]: # Create the Gradio Interface
import gradio as gr

input_fields = [gr.Number(label=col) for col in df.drop(columns=target).columns]

interface = gr.Interface(fn=predict_student_performance,
                        inputs=input_fields,
                        outputs="label",
                        title="🎓 Student Performance Predictor")

interface.launch()

```

It looks like you are running Gradio on a hosted Jupyter notebook. For the Gradio app to work, sharing must be enabled. Automatically setting `share=True` (you can turn this off by setting `share=False` in `launch()` explicitly).

Colab notebook detected. To show errors in colab notebook, set debug=True in launch()

\* Running on public URL: <https://fb88b1adb7c2729bc5.gradio.live>

This share link expires in 1 week. For free permanent hosting and GPU upgrades, run `gradio deploy` from the terminal in the working directory to deploy to Hugging Face Spaces (<https://huggingface.co/spaces>)





# Student Performance Predictor

distance\_from\_home

distance\_from\_last\_transaction

ratio\_to\_median\_purchase\_price

repeat\_retailer

used\_chip

Out[26]: