Module_3:

Team Members:

Esther Akinade & Alexander Kremsreiter

Project Title:

OncoPredict

Project Goal:

This project seeks to see how different gene types, CDH1, OCLN, SNAI1, SNAI2, and VIM, and there levels and accurate prediction to see if metastasis would actually occur a known type of metastatic cancer, Rectum Adenocarcinoma.

Disease Background:

- Cancer hallmark focus: Tissue Invasion and Metastasis
- Overview of hallmark: This hallmark refers to the ability of cancer cells to detach from the original tumor, invade surrounding tissue, enter and travel througn the bloodstream and lymphatic system, and create new tumors in other parts of the body.
- Genes associated with hallmark to be studied (describe the role of each gene, signaling pathway, or gene set you are going to investigate): The genes VIM that we will try to investiage would be the CDH1, OCLN, SNAI1, SNAI2, and VIM genes. CDH1 encodes for E-adherin that maintains cell adhesion. With the loss of CDH1 cell adhesion is disrupted and it allows tumor cells to detach and migrate. OCLN, which is Occludin that encodes for the prevelent of tight junctions and helps maintain the epithelial barriers. SNAI1, Snail, encodes for the repression of CDH1 genes in cancer. It also drives the motilitiy and invasion of the cancer cells. SNAI2(Slug) promotes resistance to apoptosis during EMT and represses epithelial genes. Lastly, VIM, vimentin, is used to provide structure and support to cell shape, adhesion, and motility. In cancer, it helps with cell migration, and the invasion of other tissues.

Will you be focusing on a single cancer type or looking across cancer types? Depending on your decision, update this section to include relevant information about the disease at the appropriate level of detail. Regardless, each bullet point should be filled in. If you are looking at multiple cancer types, you should investigate differences between the types (e.g. what is the most prevalent cancer type? What type has the highest mortality rate?) and similarities (e.g. what sorts of treatments exist across the board for cancer patients? what is common to all cancers in terms of biological mechanisms?). Note that this is a smaller list than the initial 11 in Module 1.

Currently, we plan on researching Rectum Adenocarcinoma(Rectal and Colon Cancer).

- Prevalence & incidence:https://www.cancer.org/cancer/survivorship/cancer-prevalence.html https://www.cancer.gov/types/common-cancers

- Cancer prevalence is about 18.6 million people that are living in the United States. The most prevalent cancer type that we have chosen is breast cancer. In 2025, the incidence for all cancer is about 2,041,910 new cancer cases. The incidence for breast cancer is 316,950. Additionally, we are focusing on Rectum Adenocarcinoma (Rectal and Colon Canacer) prevelence in males. So far in, 2025 there are 154,270 estminated new cases and 52,900 estimated deaths.

- Risk factors (genetic, lifestyle) & Societal determinants https://www.cancer.gov/about-cancer/causes-prevention/risk Risk factors that are included as age, alchol, cancer-causing substance, chronic inflamation, diet, hormones, immunosuppression, infectious agents, obesity, radiation, sunlight, and tobacco.

- Standard of care treatments (& reimbursement) https://jnccn.org/configurable/content/journals$002fjnccn$002f22$002f5$002farticle-p331.xml?t:ac=journals%24002fjnccn%24002f22%24002f5%24002farticle-p331.xml The standard of care for treatment include surgery for removal, radiation, chemotheraphy, targeted molecular therapies, and immunotherapy.

- Biological mechanisms (anatomy, organ physiology, cell & molecular physiology)https://geoniti.com/articles/cancer-cell-development-mechanisms-implications/ Breast cancer spreads through the axillary lymph nodes into the bloodstreams, then to the bone, liver, lungs and brain. In prostate cancer, it quickly metastasizes to bone. The key mechanism in cancer development include the genetic mutation that disrupts the normal cellular function. These mutation leads to uncontrolled cell proliferation. Additionally, cancer cells are able to negate the normal signalling pathways and promote their own growth and survival.

https://my.clevelandclinic.org/health/diseases/21733-rectal-cancer, https://www.mayoclinic.org/diseases-conditions/colon-cancer/symptoms-causes/syc-20353669?cjdata=MXxOfDB8WXww&cjevent=41c575c3bd0011f0816a032c0a82b836&cm_mmc=CJ-

Rectal cancer develops slowly in the inner lining of the rectum (the last several inches of the large intestine). It begins as adenomas, clumps of abnormal cells(polyps). These polyps develop from the glandular epithelial cells that secrete mucus that facilitates the passing of stool, but these cells can grow abnormally which can turn into adenocarcinoma. Yet it may take ten to fifteen years for the adenomas to turn into tumor masses. Rectal cancer can be deteced primarily from a Colonoscopy to check for suspicious tissue. This cancer can also metastasize by invading deeper layers beyond the inner lining of the large intestine. These layers include the submucosa then the muscular wall and finally the outer layers and nearby tissue. Then the cancerous cells can access the blood stream and lymphatic system to the rest of the body.

Data-Set:

We found a binary measure of metastasis to be most revelant to our question at hand. Pathological Metastasis Status (M) shows tumor staging as a binary unit, M0 = no metastsis and M1 = metastasis. For our research data concerning the CDH1, OCLN, SNAI1, SNAI2, and VIM

genes would be extremely useful for understanding trends within Metastasis. We want to relate the levels of the genes with the binary measure of metastasis to correlate how the differe gene levels can predict possibility of metastasis. If we decide to follow this gene to answer our question then we will need to utilize outside resources or change the approach in how we look at our data.

This data includes combined data sets from 11,160 patients across 33 different cancer types from 161 hospitals/tissue banks. The purpose is to link survival analyses to genomics. The dataset measures four major clinical outcome endpoints: OS (overall survival), PFI (progression-free interval), DFI(disease-free interval), and DSS (disease-specific survival). Since there is a wide vairety of data categories there is not just one type of unit. Some categories use an area of the body or a type of cancer (string) as a result, other are binary to answer a yes/no question, and some use integers as measurements, so there is not one standard unit across the enire data set.

Data Analyis:

Methods

The machine learning technique I am using is classification using the Logistic Regression because it allows us to test whether differences in the gene expression levels in CDH1, OCLN, SNAI1, SNAI2, and VIM are associated with the possiblity of metastasis. Logisitc Regression is the best choice for this problem type because the outcome variable is binary; either M0= no metastasis or M1 = Metastasis.

*What is this method optimizing? How does the model decide it is "good enough"?*Logistic Regression optimizes the log-likelihood function, which measures how well the model's predicted outcome maches the actual outcome. In our model, it is measuring how well the predicted possiblites matches the actual outcome of metastasis. The model decided it is "good enough" when the loss function converges, which means that the loss fuction falls below a threshold.

Analysis

Create a filtered CSV file of relevant data: the genes of interest (CDH1, OCLN, SNAI1, SNAI2, and VIM) with their corresponding patient barcodes and additional respective data.

```
import pandas as pd

#1. Load the dataset
# Replace with your own path if needed
file_path = "GSE62944_subsample_log2TPM.csv"
df = pd.read_csv(file_path)

#2. Define genes of interest
genes_of_interest = ["CDH1", "OCLN", "VIM", "SNAI1", "SNAI2"]

#3. Filter dataset for those genes
# The dataset has genes as rows, samples (barcodes) as columns
subset_df = df[df["Unnamed: 0"].isin(genes_of_interest)].copy()
```

```
# Rename the gene column for clarity
subset_df.rename(columns={"Unnamed: 0": "Gene"}, inplace=True)

# 4. Transpose so that barcodes are rows and genes are columns
subset_df = subset_df.set_index("Gene").T.reset_index()

# Rename the first column to indicate it contains barcodes
subset_df.rename(columns={"index": "Barcode"}, inplace=True)

#5. Save to new CSV
output_path = "EMT_gene_expression_subset.csv"
subset_df.to_csv(output_path, index=False)

print(f"Filtered dataset saved to: {output_path}")
```

```
Filtered dataset saved to: EMT_gene_expression_subset.csv
```

Loads filtered data set and meta data to further create another csv with information on the cancer type associated with each patient

```
import pandas as pd

#1. Load both datasets
emt_df = pd.read_csv("EMT_gene_expression_subset.csv")
meta_df = pd.read_csv("GSE62944_metadata.csv")

#2. Merge on the barcode/sample
# 'Barcode' in EMT file corresponds to 'sample' in metadata
merged_df = pd.merge(
    emt_df,
    meta_df[["sample", "cancer_type"]],
    left_on="Barcode",
    right_on="sample",
    how="left"
)

# Drop redundant column
merged_df.drop(columns=["sample"], inplace=True)

#3. Save the merged file
merged_df.to_csv("EMT_gene_expression_with_cancer_type.csv",
index=False)

print("Merged file saved as EMT_gene_expression_with_cancer_type.csv")
```

```
Merged file saved as EMT_gene_expression_with_cancer_type.csv
```

Filters the filtered CSV and meta data to find just the patients with cancer types of interest.

```python
import pandas as pd

#1. Load both CSV files
genes_df = pd.read_csv("EMT_gene_expression_with_cancer_type.csv")
metadata_df = pd.read_csv("GSE62944_metadata.csv")

#2. Filter for BRCA samples
brca_df = genes_df[genes_df["cancer_type"].str.upper() == "READ"]

#3. Merge using Barcode (from genes_df) and sample (from metadata_df)
merged_df = pd.merge(
    brca_df,
    metadata_df[["sample", "ajcc_metastasis_pathologic_pm"]],
    left_on="Barcode",
    right_on="sample",
    how="left"
)

#4. Optionally, drop the extra 'sample' column (since Barcode is
already there)
merged_df = merged_df.drop(columns=["sample"])

#5. Save the result
merged_df.to_csv("GENES_READ.csv", index=False)

print(" GENES_READ.csv created successfully with
ajcc_metastasis_pathologic_pm added")
```

```
 GENES_READ.csv created successfully with
ajcc_metastasis_pathologic_pm added
```

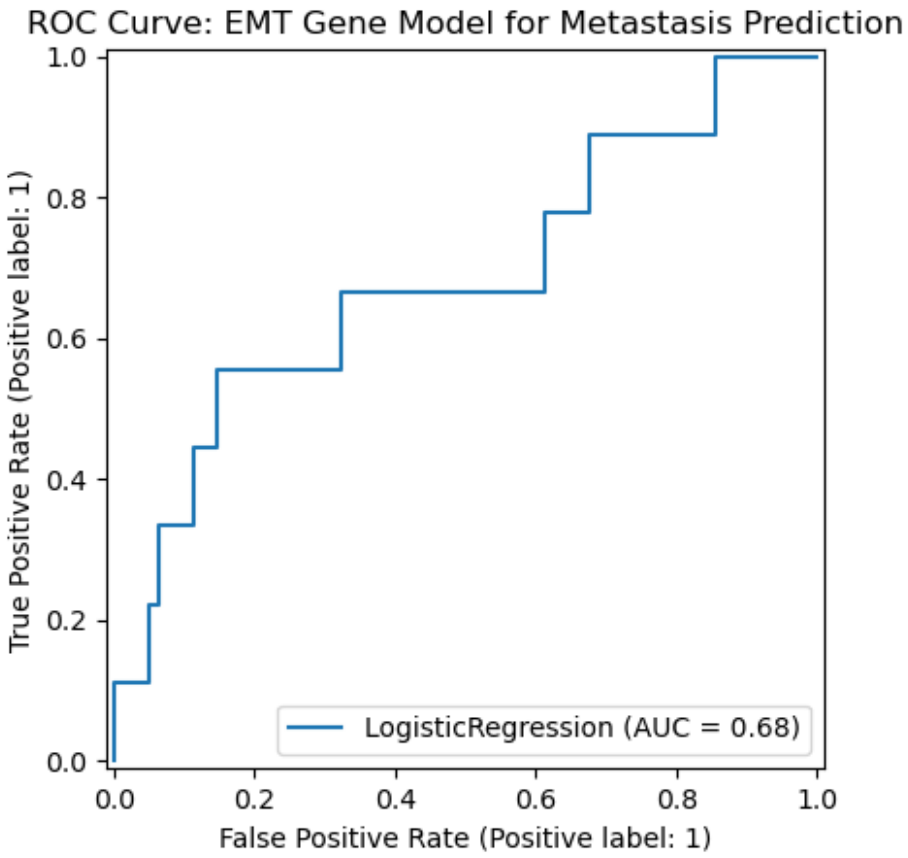
```python
import pandas as pd
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import accuracy_score, roc_auc_score,
confusion_matrix, classification_report, RocCurveDisplay
import matplotlib.pyplot as plt

# Load dataset
df = pd.read_csv("GENES_READ.csv")

# Define target and EMT-related genes
target_col = "ajcc_metastasis_pathologic_pm"
genes = ["CDH1", "OCLN", "SNAI1", "SNAI2", "VIM"]

# Keep only columns that exist in your file
genes = [g for g in genes if g in df.columns]
df = df.dropna(subset=genes + [target_col])

# Keep only rows that are M0 or M1
df = df[df[target_col].isin(["M0", "M1"])]
```

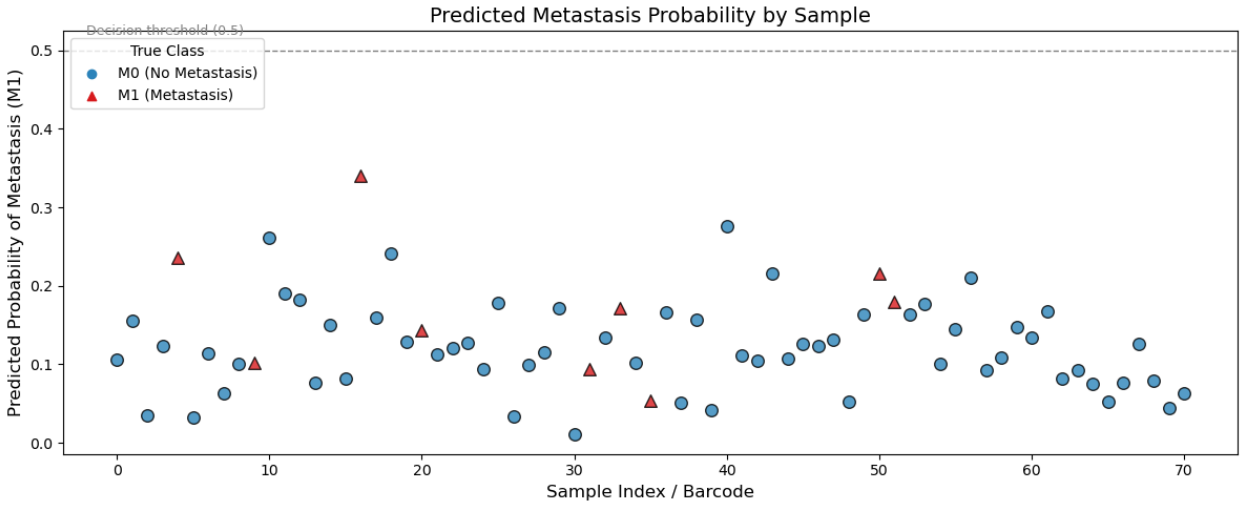
```
# Encode metastasis (M0 = 0, M1 = 1)
le = LabelEncoder()
df["Metastasis_binary"] = le.fit_transform(df[target_col])

# Prepare features (X) and target (y)
X = df[genes]
y = df["Metastasis_binary"]

# --- Fit logistic regression ---
model = LogisticRegression(max_iter=1000)
model.fit(X, y)

# Predictions and probabilities
y_pred = model.predict(X)
y_prob = model.predict_proba(X)[:, 1]

# --- Performance metrics ---
acc = accuracy_score(y, y_pred)
try:
    auc = roc_auc_score(y, y_prob)
except ValueError:
    auc = None

cm = confusion_matrix(y, y_pred)

print(" Logistic Regression on EMT Genes and Metastasis\n" + "-" * 55)
print(f"Accuracy: {acc:.3f}")
print(f"AUC (ROC): {auc:.3f}" if auc is not None else "AUC could not
be computed.")
print("\nConfusion Matrix:")
print(cm)
print("\nClassification Report:")
print(classification_report(y, y_pred))

# --- Coefficients ---
print("Gene Coefficients:")
for gene, coef in zip(genes, model.coef_[0]):
    direction = "↑ (positive correlation with metastasis)" if coef > 0
else "↓ (negative correlation)"
    print(f"{gene:10s}: {coef: .4f}  {direction}")

# --- ROC Curve ---
if len(set(y)) == 2:
    RocCurveDisplay.from_estimator(model, X, y)
    plt.title("ROC Curve: EMT Gene Model for Metastasis Prediction")
    plt.show()

# --- Probability plot color + shape by M0/M1 ---
plt.figure(figsize=(12, 5))
```

```python
# Define colors and markers
colors = ['#2b83ba' if label == 0 else '#d7191c' for label in y]  # blue=M0, red=M1
markers = ['o' if label == 0 else '^' for label in y]  # circle for M0, triangle for M1

# Scatter plot with different shapes
for i, (prob, label) in enumerate(zip(y_prob, y)):
    plt.scatter(i, prob,
                color=colors[i],
                marker=markers[i],
                s=70,
                edgecolor='k',
                alpha=0.8)

plt.axhline(0.5, color='gray', linestyle='--', linewidth=1)
plt.text(-2, 0.52, "Decision threshold (0.5)", fontsize=9, color='gray')

plt.xlabel("Sample Index / Barcode", fontsize=12)
plt.ylabel("Predicted Probability of Metastasis (M1)", fontsize=12)
plt.title("Predicted Metastasis Probability by Sample", fontsize=14)

# Legend
plt.scatter([], [], color='#2b83ba', marker='o', label='M0 (No Metastasis)')
plt.scatter([], [], color='#d7191c', marker='^', label='M1 (Metastasis)')
plt.legend(title="True Class", loc="upper left")

plt.tight_layout()
plt.show()
```

```
 Logistic Regression on EMT Genes and Metastasis
--------------------------------------------------------
Accuracy: 0.873
AUC (ROC): 0.685

Confusion Matrix:
[[62  0]
 [ 9  0]]

Classification Report:
              precision    recall  f1-score   support

           0       0.87      1.00      0.93        62
           1       0.00      0.00      0.00         9

    accuracy                           0.87        71
```

```
   macro avg       0.44      0.50      0.47        71
weighted avg       0.76      0.87      0.81        71

Gene Coefficients:
CDH1      :  0.0679  ↑ (positive correlation with metastasis)
OCLN      :  0.1575  ↑ (positive correlation with metastasis)
SNAI1     :  0.5313  ↑ (positive correlation with metastasis)
SNAI2     : -0.2677  ↓ (negative correlation)
VIM       :  0.5072  ↑ (positive correlation with metastasis)

c:\Users\esthe\anaconda3\Lib\site-packages\sklearn\metrics\
_classification.py:1565: UndefinedMetricWarning: Precision is ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, f"{metric.capitalize()} is",
len(result))
c:\Users\esthe\anaconda3\Lib\site-packages\sklearn\metrics\
_classification.py:1565: UndefinedMetricWarning: Precision is ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, f"{metric.capitalize()} is",
len(result))
c:\Users\esthe\anaconda3\Lib\site-packages\sklearn\metrics\
_classification.py:1565: UndefinedMetricWarning: Precision is ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, f"{metric.capitalize()} is",
len(result))
```





Verify and validate your analysis:

The specific method that I chose to determine how well my model is preforming is the ROC-AUC scoure. This score measure how well the logistic regression can distiguish between the M0 and M1. If the AUc score is equal to 1, it is a perfect prediction and if it is less than 0.5, it is worse than random prediction.

To verify my analysis is by spliting the dataset into training and test sets, which ensures that the logistic regression was not has not ben doen before. Additionally, it used ROC-AUC rather than just accuracy because it has a better ability on distigusing and ranking.

Based on other published work, EMT markers such as CDH1, SNAI1, SNAI2, and VIM play important roles in promoting tissue invasion and metastasis. For example, studies have shown that EMT-related genes can provide expression patterns that help facilitate the prognosis of metastatic cancers. However, my AUC-ROC score was 0.351, indicating that the EMT genes selected in this analysis did not perform well at predicting metastasis. It is also important to note that there are other EMT-related genes and molecular factors that likely contribute to cancer prognosis and should be considered in future models.

```
import pandas as pd
import numpy as np
from sklearn.linear_model import LogisticRegression
from sklearn.dummy import DummyClassifier
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.metrics import (
    accuracy_score, confusion_matrix, classification_report,
    roc_auc_score, ConfusionMatrixDisplay, RocCurveDisplay
)
import matplotlib.pyplot as plt
import seaborn as sns

# 1 LOAD & PREPARE DATA
df = pd.read_csv("GENES_READ.csv")

target_col = "ajcc_metastasis_pathologic_pm"
genes = ["CDH1", "OCLN", "SNAI1", "SNAI2", "VIM"]

# Keep only existing columns
genes = [g for g in genes if g in df.columns]
df = df.dropna(subset=genes + [target_col])

# Filter to M0 / M1 only
df = df[df[target_col].isin(["M0", "M1"])]

# Encode metastasis (M0=0, M1=1)
df["Metastasis_binary"] = df[target_col].map({"M0": 0, "M1": 1})

X = df[genes]
y = df["Metastasis_binary"]

print("=" * 60)
print("DATA SUMMARY")
print("=" * 60)
print(f"Samples: {len(df)}")
print(f"M0: {(y==0).sum()}  |  M1: {(y==1).sum()}")
print(f"Features used: {len(genes)} ({', '.join(genes)})")
```

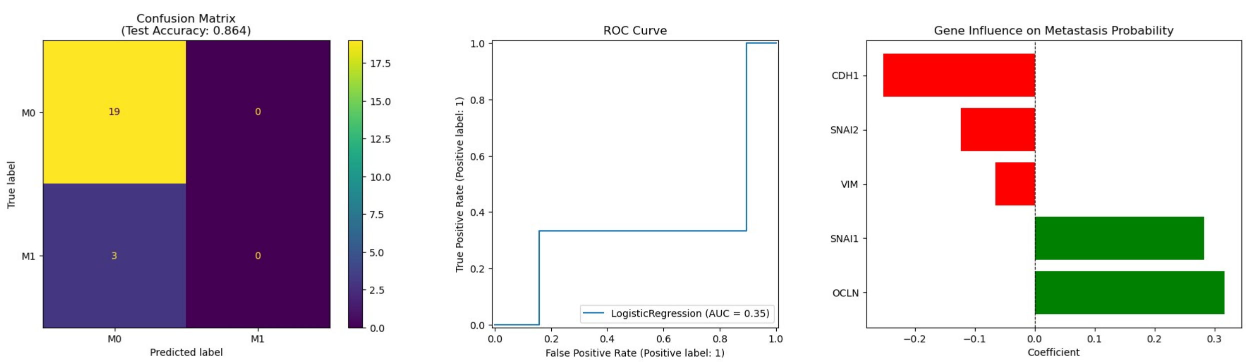
```
#
# 2 TRAIN/TEST SPLIT
#
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, stratify=y, random_state=42
)

print(f"\nTrain samples: {len(X_train)} | Test samples:
{len(X_test)}")

#
# 3 TRAIN LOGISTIC REGRESSION MODEL
#
logreg = LogisticRegression(penalty='l2', C=1.0, max_iter=1000,
random_state=42)
logreg.fit(X_train, y_train)

# Predictions
y_pred_train = logreg.predict(X_train)
y_pred_test = logreg.predict(X_test)
y_prob_test = logreg.predict_proba(X_test)[:, 1]

# 4 DUMMY BASELINE COMPARISON
dummy = DummyClassifier(strategy="most_frequent", random_state=42)
dummy.fit(X_train, y_train)
dummy_pred = dummy.predict(X_test)

# 5 EVALUATION

train_acc = accuracy_score(y_train, y_pred_train)
test_acc = accuracy_score(y_test, y_pred_test)
dummy_acc = accuracy_score(y_test, dummy_pred)
auc = roc_auc_score(y_test, y_prob_test)

print("\n" + "=" * 60)
print("MODEL PERFORMANCE")
print("=" * 60)
print(f"Training accuracy:  {train_acc:.3f}")
print(f"Test accuracy:      {test_acc:.3f}")
print(f"Dummy baseline:     {dummy_acc:.3f}")
print(f"Improvement vs dummy: {test_acc - dummy_acc:+.3f}")
print(f"AUC-ROC:            {auc:.3f}")

# Confusion matrix
cm = confusion_matrix(y_test, y_pred_test)
print("\nConfusion Matrix:\n", cm)
print("\nClassification Report:\n")
print(classification_report(y_test, y_pred_test, target_names=["M0",
"M1"]))
```

```python
# 6 CROSS-VALIDATION
cv_scores = cross_val_score(logreg, X, y, cv=5, scoring='accuracy')
print("=" * 60)
print("5-FOLD CROSS-VALIDATION")
print("=" * 60)
print(f"CV Accuracy: {cv_scores.mean():.3f} ± {cv_scores.std():.3f}")
print(f"Individual folds: {np.round(cv_scores, 3)}")

# 7 GENE COEFFICIENTS (Interpretation)
coef_df = pd.DataFrame({
    'Gene': genes,
    'Coefficient': logreg.coef_[0]
}).sort_values('Coefficient', ascending=False)

print("\n" + "=" * 60)
print("GENE COEFFICIENTS (INTERPRETATION)")
print("=" * 60)
print("Positive = higher expression → higher metastasis probability
(M1)")
print("Negative = higher expression → lower metastasis probability
(M0)\n")

for _, row in coef_df.iterrows():
    arrow = "↑" if row["Coefficient"] > 0 else "↓"
    print(f"{row['Gene']:10s}: {row['Coefficient']:+.4f}  {arrow}")

# 8 VISUALIZATIONS
fig, axes = plt.subplots(1, 3, figsize=(18, 5))

# 1. Confusion Matrix
ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=["M0",
"M1"]).plot(ax=axes[0])
axes[0].set_title(f"Confusion Matrix\n(Test Accuracy:
{test_acc:.3f})")

# 2. ROC Curve
RocCurveDisplay.from_estimator(logreg, X_test, y_test, ax=axes[1])
axes[1].set_title("ROC Curve")

# 3. Gene Coefficients
colors = ['green' if c > 0 else 'red' for c in coef_df["Coefficient"]]
axes[2].barh(coef_df["Gene"], coef_df["Coefficient"], color=colors)
axes[2].set_xlabel("Coefficient")
axes[2].set_title("Gene Influence on Metastasis Probability")
axes[2].axvline(0, color='black', linestyle='--', linewidth=0.8)

plt.tight_layout()
plt.show()

print("Original genes list:", genes)
```

```
print("Available columns in CSV:", df.columns.tolist())
print("Missing genes:", [g for g in genes if g not in df.columns])
```

```
============================================================
DATA SUMMARY
============================================================
Samples: 71
M0: 62  |  M1: 9
Features used: 5 (CDH1, OCLN, SNAI1, SNAI2, VIM)

Train samples: 49 | Test samples: 22

============================================================
MODEL PERFORMANCE
============================================================
Training accuracy:  0.878
Test accuracy:      0.864
Dummy baseline:     0.864
Improvement vs dummy: +0.000
AUC-ROC:            0.351

Confusion Matrix:
 [[19  0]
 [ 3  0]]

Classification Report:

              precision    recall  f1-score   support

          M0       0.86      1.00      0.93        19
          M1       0.00      0.00      0.00         3

    accuracy                           0.86        22
   macro avg       0.43      0.50      0.46        22
weighted avg       0.75      0.86      0.80        22

============================================================
5-FOLD CROSS-VALIDATION
============================================================
CV Accuracy: 0.873 ± 0.028
Individual folds: [0.867 0.929 0.857 0.857 0.857]

============================================================
GENE COEFFICIENTS (INTERPRETATION)
============================================================
Positive = higher expression → higher metastasis probability (M1)
Negative = higher expression → lower metastasis probability (M0)

OCLN      : +0.3169  ↑
SNAI1     : +0.2823  ↑
```

```
VIM       : -0.0652  ↓
SNAI2     : -0.1226  ↓
CDH1      : -0.2529  ↓

c:\Users\esthe\anaconda3\Lib\site-packages\sklearn\metrics\
_classification.py:1565: UndefinedMetricWarning: Precision is ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, f"{metric.capitalize()} is",
len(result))
c:\Users\esthe\anaconda3\Lib\site-packages\sklearn\metrics\
_classification.py:1565: UndefinedMetricWarning: Precision is ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, f"{metric.capitalize()} is",
len(result))
c:\Users\esthe\anaconda3\Lib\site-packages\sklearn\metrics\
_classification.py:1565: UndefinedMetricWarning: Precision is ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, f"{metric.capitalize()} is",
len(result))
```




```
Original genes list: ['CDH1', 'OCLN', 'SNAI1', 'SNAI2', 'VIM']
Available columns in CSV: ['Barcode', 'CDH1', 'OCLN', 'SNAI1',
'SNAI2', 'VIM', 'cancer_type', 'ajcc_metastasis_pathologic_pm',
'Metastasis_binary']
Missing genes: []
```

Conclusions and Ethical Implications:

Based on our data, AUC score = 0.351, which means that it is below random guessing, we found that logistic regression could not accurately or reliably predict metastasis using the five genes: CDH1, OCLN, SNAI1, SNAI2, and VIM. The primary reason for this failure is the lack of metastatic (M1) data. This is shown in our confusion matrixs that had about 19 M0 and 3 M1.

As a result, our model cannot ethically support many clinical conclusions, because it would likely produce false positives and false negatives if applied to real patients. Such errors could lead to

inappropriate treatment decisions, delayed interventions, or unnecessary emotional and medical burden for patients. However, one concluison that was supported was that SNAI1 gene increased as M1 increased. Overall, this data shows that more genes are significant to the metastasis hallmark.

Additionally, the structure of the dataset itself may contribute to this limitation. Because TCGA and similar datasets enforce strong privacy safeguards, it is possible that patients with confirmed metastatic cancer were underrepresented and withdrew consent, leading to fewer available M1 samples. This reinforces the importance of recognizing dataset limitations and avoiding overinterpretation of biased or incomplete clinical data.

Limitations and Future Work:

For our question of predicting whether a patient will suffer from metastasis or not, our model required a high amount of data points from both patients with and without metastatic cancer. Our data set had an overwhelming amount of patients without any data in the binary category (M0 for no metastasis and M1 for metastasis) for metastatic cancer. Additionally, our data set had very little patients who actually suffer from metastatic cancer. This further limited our model as we focused on five genes overall, meaning we filtered out many data points.

In future iterations of this project we would ideally have more data dealing with positive metastatic cancer results. Obviously we do not want people to suffer from metastaic cancers but with a lack of data points we cannot make accurate predictions. Additionally, we would focus on 10-20 genes rather than 5 to draw further correlation between gene levels and metastatic cancer status within patients. In the future we would consider using a nonlinear model to understand correlations between data. Additionally, with any results found we would want to utilize additional data sets to further prove our findings. Another future work would be not doing retraining our model on the split data; instead we would train the test data on the original model.

NOTES FROM YOUR TEAM:

To begin, we are seperating our data to see if we are able to actually work with the data that we have. We are working on using our first data set, breast cancer, and finding the CDH1 levels that are associated with it. This is found in of analysis section of sepreating our data set.

We changed what we are researching and the different genes that contribute to it, so the Background infromation might not perfectly if what is found in the overall notbook.

QUESTIONS FOR YOUR TA: