

02_explain_local

June 30, 2025

0.1 Imports

```
[55]: import shap, joblib
import os
import matplotlib.pyplot as plt
from pathlib import Path

# find project root by climbing up until README is found
root = Path.cwd()
while not (root / "README.md").exists():
    root = root.parent
os.chdir(root)          # now CWD is project root
os.makedirs("reports", exist_ok=True)
from credit_risk.data_ingest import load_data, clean_data
from credit_risk.features import generate_features
```

1 Ingest, clean, featurize

```
[56]: df_raw = load_data("data/raw/borrowers.csv")
df_clean = clean_data(df_raw)
x_train, x_test, y_train, y_test = generate_features(df, stratify=True)

print(f"Loaded {len(df_raw)} rows -> {len(x_train)} train / {len(x_test)} test_
↳samples")
```

```
Loaded 5 rows and 6 columns from data/raw/borrowers.csv
Dropped 0 duplicate rows
Filled missing 'employment_length' with median=7.0
Filled missing 'age' with median=45.0
Filled missing 'annual_income' with median=60000.0
Clipped 'debt_to_income' to [0,1]
One-hot encoded columns: ['emp_bin', 'age_bin']
Cleaned data has 5 rows and 16 columns
Feature generation complete: 4 training samples, 1 test samples
Scaled numeric columns: ['annual_income', 'employment_length', 'credit_score',
'age', 'debt_to_income', 'dti_pct']
Stratification: on
Loaded 5 rows -> 4 train / 1 test samples
```

```
/Users/glennasher/credit-risk-project/src/credit_risk/features.py:43:
UserWarning: Stratified split failed (The test_size = 1 should be greater or
equal to the number of classes = 2); falling back to unstratified split.
warnings.warn(
```

1.1 Load best model

```
[57]: model = joblib.load("models/best_model.joblib")
```

1.2 Build small background dataset for SHAP

```
[58]: N_bg = min(len(x_train), 50)
background = x_train.sample(N_bg, replace=(N_bg>len(x_train)), random_state=0)

# coerce to numpy arrays of floats only
xb = background.values.astype(float)
xt = x_test.values.astype(float)

# Use Linear Explainer (supports linear/logistic models)
explainer = shap.LinearExplainer(
    model,
    xb,
    feature_perturbation = "interventional"      # recommended for tabular
)

# Compute SHAP values on test set
# shap_values is a matrix (n_samples x n_features)
shap_values = explainer.shap_values(xt)
```

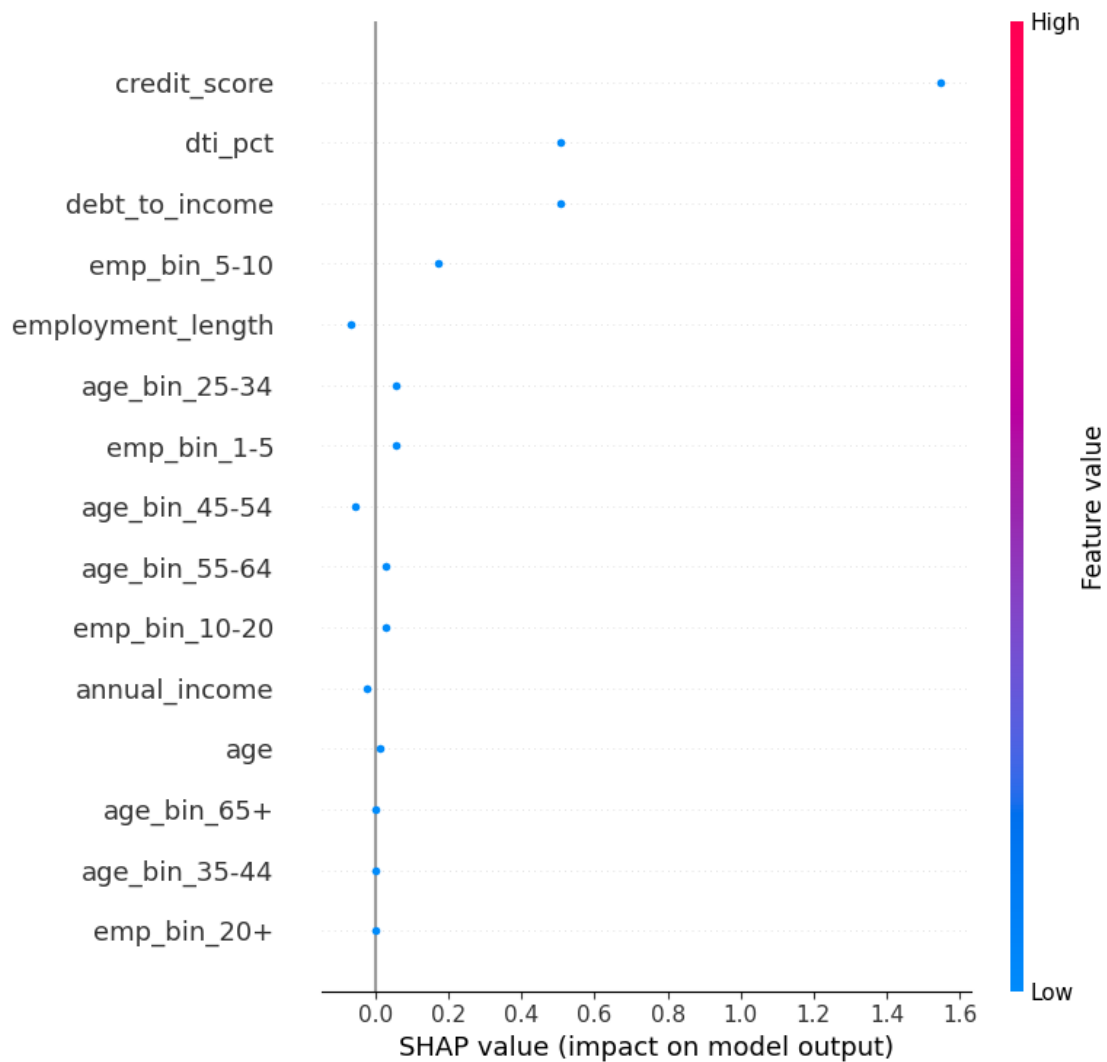
```
/Users/glennasher/credit-risk-project/.venv/lib/python3.13/site-
packages/shap/explainers/_linear.py:99: FutureWarning: The feature_perturbation
option is now deprecated in favor of using the appropriate masker
(maskers.Independent, maskers.Partition or maskers.Impute).
warnings.warn(wmsg, FutureWarning)
```

1.3 Plot a summary plot (Global feature importance)

```
[59]: plt.figure(figsize=(8,6))
shap.summary_plot(
    shap_values,
    xt,
    feature_names = x_test.columns
)
```

```
/var/folders/z8/jsh_8yzd3l51rqktyzb1cdd80000gn/T/ipykernel_2693/3984942613.py:2:
FutureWarning: The NumPy global RNG was seeded by calling `np.random.seed`. In a
future version this function will no longer use the global RNG. Pass `rng`
```

explicitly to opt-in to the new behaviour and silence this warning.
shap.summary_plot(

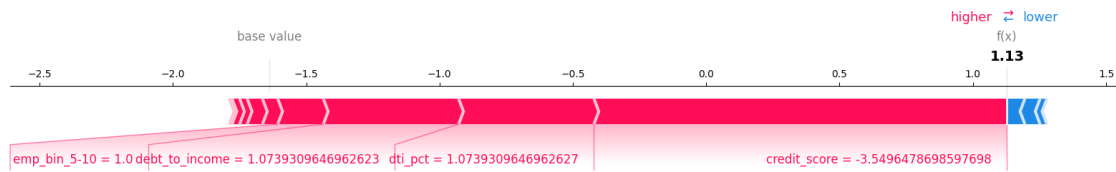


1.4 Force plot for a single example (inline)

```
[60]: idx = 0      # change to look at other rows
shap.initjs()
force_plot = shap.force_plot(
    explainer.expected_value,
    shap_values[idx,:],
    xt[idx,:),
    feature_names = x_test.columns,
    matplotlib = True
)
```

```
plt.savefig("reports/shap_force_0.png", bbox_inches="tight")
```

<IPython.core.display.HTML object>



<Figure size 640x480 with 0 Axes>

1.5 Saving a static PNG

```
[61]: import matplotlib.pyplot as plt
plt.tight_layout()
plt.savefig("reports/shap_force_example.png", dpi=150)
```

<Figure size 640x480 with 0 Axes>