

My Path Module | WorldQuant U...work-ds-curricu (2) - JupyterLabData-Science-Lab/GridSearchCV | +

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

NameLast Modified

ipynb\_chec...2 days ago

data2 months ago

images4 days ago

051-workin...4 days ago

052-imbalan...2 months ago

053-random...2 months ago

054-gradien...an hour ago

055-assign...3 minutes ago

056-data-di...2 months ago

my\_predicto...2 months ago

my\_predicto...2 months ago

055-assignment.ipynb054-gradient-boosting.ipynb

Python 3 (pykernel)

## 5.5. Bankruptcy in Taiwan tw

```
[1]: import wget_grader
      wget_grader.init("Project 5 Assessment")

[2]: # Import libraries here
      from sklearn.base import ClassifierMixin
      from sklearn.pipeline import Pipeline
      import gzip
      import json
      import pickle

      import pandas as pd
      import matplotlib.pyplot as plt
      import pandas as pd
      import seaborn as sns
      import wget_grader
      from imblearn.over_sampling import RandomOverSampler
      from imblearn.under_sampling import RandomUnderSampler
      from sklearn.metrics import SimpleImputer
      from sklearn.metrics import (
          ConfusionMatrixDisplay,
          classification_report,
          confusion_matrix,
      )

      from sklearn.pipeline import make_pipeline
      from sklearn.tree import DecisionTreeClassifier
      from sklearn.ensemble import RandomForestClassifier
      from sklearn.model_selection import GridSearchCV, cross_val_score, train_test_split
      import ipywidgets as widgets
      from ipywidgets import interact
      from sklearn.ensemble import GradientBoostingClassifier
      from teaching_tools.widgets import ConfusionMatrixWidget
```

### Prepare Data

Simple05Python 3 (pykernel) | Idle

Mode: CommandLn 1, Col 1English (United States)055-assignment.ipynb

26°C Haze

Search

ENG17-03-202323:36

My Path Module | WorldQuant U...work-ds-curricu (2) - JupyterLabData-Science-Lab/GridSearchCV | +

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

NameLast Modified

ipynb\_chec...2 days ago

data2 months ago

images4 days ago

051-workin...4 days ago

052-imbalan...2 months ago

053-random...2 months ago

054-gradien...an hour ago

055-assign...3 minutes ago

056-data-di...2 months ago

my\_predicto...2 months ago

my\_predicto...2 months ago

055-assignment.ipynb054-gradient-boosting.ipynb

Python 3 (pykernel)

## Prepare Data

### Import

**Task 5.5.1:** Load the contents of the "data/taiwan-bankruptcy-data.json.gz" and assign it to the variable 'taiwan\_data'.  
Note that 'taiwan\_data' should be a dictionary. You'll create a DataFrame in a later task.

```
[3]: # Load data file
      with gzip.open("data/taiwan-bankruptcy-data.json.gz", "r") as f:
          taiwan_data = json.load(f)

      print(type(taiwan_data))
      <class 'dict'>

[4]: wget_grader.grade("Project 5 Assessment", "Task 5.5.1", taiwan_data["metadata"])



✓



That's the right answer. Keep it up!



Score: 1



Task 5.5.2: Extract the key names from 'taiwan_data' and assign them to the variable 'taiwan_data_keys'.



Tip: The data in this assignment might be organized differently than the data from the project, so be sure to inspect it first.



```
[5]: taiwan_data_keys = taiwan_data.keys()
      print(taiwan_data_keys)

      dict_keys(['schema', 'metadata', 'observations'])
```


```

Simple05Python 3 (pykernel) | Idle

Mode: CommandLn 1, Col 1English (United States)055-assignment.ipynb

26°C Haze

Search

ENG17-03-202323:36

My Path Module | WorldQuant U x work/ds-curricu (2) - JupyterLab x Data-Science-Lab/GridSearchCV x +

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- .ipynb\_chec... 2 days ago
- data 2 months ago
- images 4 days ago
- 051-workin... 4 days ago
- 052-imbalan... 2 months ago
- 053-random... 2 months ago
- 054-gradien... an hour ago
- 055-assign... 4 minutes ago
- 056-data-di... 2 months ago
- my\_predicto... 2 months ago
- my\_predicto... 2 months ago

055-assignment.ipynb

```
[5]: taiwan_data_keys = taiwan_data.keys()
    print(taiwan_data_keys)

dict_keys(['schema', 'metadata', 'observations'])

[7]: wqet_grader.grade("Project 5 Assessment", "Task 5.5.2", list(taiwan_data_keys))

You're making this look easy.
Score: 1

Task 5.5.3: Calculate how many companies are in 'taiwan_data' and assign the result to 'n_companies'.

[8]: len(taiwan_data["observations"])

[8]: 6137

[9]: n_companies = len(taiwan_data["observations"])
    print(n_companies)

6137

[10]: wqet_grader.grade("Project 5 Assessment", "Task 5.5.3", [n_companies])

Very impressive.
Score: 1

Task 5.5.4: Calculate the number of features associated with each company and assign the result to 'n_features'.

[11]: n_features = len(taiwan_data["observations"][0])
    print(n_features)

97
```

Simple 0 5 Python 3 (ipykernel) Idle Mode: Command Ln 1, Col 1 English (United States) 055-assignment.ipynb

26°C Haze

My Path Module | WorldQuant U x work/ds-curricu (2) - JupyterLab x Data-Science-Lab/GridSearchCV x +

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- .ipynb\_chec... 2 days ago
- data 2 months ago
- images 4 days ago
- 051-workin... 4 days ago
- 052-imbalan... 2 months ago
- 053-random... 2 months ago
- 054-gradien... an hour ago
- 055-assign... 4 minutes ago
- 056-data-di... 2 months ago
- my\_predicto... 2 months ago
- my\_predicto... 2 months ago

055-assignment.ipynb

```
Task 5.5.4: Calculate the number of features associated with each company and assign the result to 'n_features'.

[11]: n_features = len(taiwan_data["observations"][0])
    print(n_features)

97

[13]: wqet_grader.grade("Project 5 Assessment", "Task 5.5.4", [n_features])

Yup. You got it.
Score: 1

Task 5.5.5: Create a 'unrangle' function that takes as input the path of a compressed JSON file and returns the file's contents as a DataFrame. Be sure that the index of the DataFrame contains the ID of the companies. When your function is complete, use it to load the data into the DataFrame 'df'.

[14]: def unrangle(filePath):
    # Open compressed file, load to dict
    with gzip.open(filePath, "r") as f:
        data = json.load(f)

    # Dictionary -> DataFrame, set index
    df = pd.DataFrame().from_dict(data["observations"]).set_index("id")

    return df

[15]: df = unrangle('data/taiwan-bankruptcy-data.json.gz')
    print("df shape:", df.shape)
    df.head()

df shape: (6137, 96)

[15]: bankrupt feat_1 feat_2 feat_3 feat_4 feat_5 feat_6 feat_7 feat_8 feat_9 ... feat_86 feat_87 feat_88 feat_89 feat_90 feat_91 feat_92 feat_93 feat_94 feat_95
id
1 True 0.370594 0.424389 0.405750 0.601457 0.601457 0.998969 0.796887 0.808809 0.302646 ... 0.716845 0.009219 0.622879 0.601453 0.827890 0.290202 0.026601 0.564050 1 0.016469
2 True 0.464291 0.538214 0.516730 0.610235 0.610235 0.998969 0.797380 0.809301 0.303556 ... 0.795297 0.008323 0.623652 0.610237 0.839969 0.283846 0.264577 0.570175 1 0.020794
```

Simple 0 5 Python 3 (ipykernel) Idle Mode: Command Ln 1, Col 1 English (United States) 055-assignment.ipynb

26°C Haze

My Path Module | WorldsQuant U

work/ds-curricu (2) - JupyterLab

Data-Science-Lab/GridSearchCV

+

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

055-assignment.ipynb

054-gradient-boosting.ipynb

Python 3 (pykernel)

df = wrangle("data/taiwan-bankruptcy-data.json.gz")  
print("df shape:", df.shape)  
df.head()

df shape: (6137, 96)

	bankrupt	feat_1	feat_2	feat_3	feat_4	feat_5	feat_6	feat_7	feat_8	feat_9	feat_10	feat_11	feat_12	feat_13	feat_14	feat_15	feat_16	feat_17	feat_18	feat_19	feat_20	feat_21	feat_22	feat_23	feat_24	feat_25
1	True	0.370594	0.424389	0.405750	0.601457	0.601457	0.998969	0.796887	0.808809	0.302646	...	0.716845	0.009219	0.622879	0.601453	0.827890	0.290202	0.026601	0.564050	1	0.016469					
2	True	0.464291	0.538214	0.516730	0.610235	0.610235	0.998946	0.797380	0.808901	0.303556	...	0.795297	0.008323	0.623652	0.610237	0.839969	0.263846	0.264577	0.570175	1	0.020794					
3	True	0.426071	0.499019	0.472295	0.601450	0.601364	0.998857	0.796403	0.808388	0.302035	...	0.774670	0.040003	0.623841	0.601449	0.836774	0.290189	0.026555	0.563706	1	0.016474					
4	True	0.399844	0.451265	0.457733	0.583541	0.583541	0.998700	0.796967	0.808966	0.303350	...	0.739555	0.003252	0.622929	0.583538	0.834697	0.281721	0.026697	0.564663	1	0.023982					
5	True	0.465022	0.538432	0.522298	0.598783	0.598783	0.998973	0.797366	0.809304	0.303475	...	0.795016	0.003878	0.623521	0.598782	0.839973	0.278514	0.024752	0.575617	1	0.035490					

5 rows x 96 columns

wqet\_grader.grade("Project 5 Assessment", "Task 5.5.5", df)

✓

Party time! 🎉 🎉 🎉

Score: 1

Explore

Task 5.5.6: Is there any missing data in the dataset? Create a Series where the index contains the name of the columns in `df` and the values are the number of `NaN`s in each column. Assign the result to `nans_by_col`. Neither the Series itself nor its index require a name.

nans\_by\_col = pd.Series(df.isnull().sum(), index=df.columns)  
print("nans\_by\_col shape:", nans\_by\_col.shape)  
nans\_by\_col.head()

nans\_by\_col shape: (96,)

Simple

0 5 Python 3 (pykernel) | Idle

Mode: Command Ln 1, Col 1 English (United States) 055-assignment.ipynb

26°C  
Haze

Search

23:36  
17-03-2023

My Path Module | WorldsQuant U

work/ds-curricu (2) - JupyterLab

Data-Science-Lab/GridSearchCV

+

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

055-assignment.ipynb

054-gradient-boosting.ipynb

Python 3 (pykernel)

Explore

Task 5.5.6: Is there any missing data in the dataset? Create a Series where the index contains the name of the columns in `df` and the values are the number of `NaN`s in each column. Assign the result to `nans_by_col`. Neither the Series itself nor its index require a name.

nans\_by\_col = pd.Series(df.isnull().sum(), index=df.columns)  
print("nans\_by\_col shape:", nans\_by\_col.shape)  
nans\_by\_col.head()

nans\_by\_col shape: (96,)

bankrupt	0
feat_1	0
feat_2	0
feat_3	0
feat_4	0
dtype	int64

wqet\_grader.grade("Project 5 Assessment", "Task 5.5.6", nans\_by\_col)

✓

Good work!

Score: 1

Task 5.5.7: Is the data imbalanced? Create a bar chart that shows the normalized value counts for the column `df["bankrupt"]`. Be sure to label your x-axis "Bankrupt", your y-axis "Frequency", and use the title "Class Balance".

# Plot class balance  
df["bankrupt"].value\_counts(normalize=True).plot(  
kind = "bar",  
xlabel = "Bankrupt",  
ylabel = "Frequency",  
title = "Class Balance"  
);  
# Don't delete the code below  
plt.savefig("images/5-5-7.png", dpi=150)

Simple

0 5 Python 3 (pykernel) | Idle

Mode: Command Ln 1, Col 1 English (United States) 055-assignment.ipynb

26°C  
Haze

Search

23:36  
17-03-2023

My Path Module | WorldsQuant U

work/ds-curricu (2) - JupyterLab

Data-Science-Lab/GridSearchCV

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name	Last Modified
050-bankruptcy-in-poland	2 days ago
051-workin...	2 months ago
052-imbalan...	4 days ago
053-random...	4 days ago
054-gradien...	2 months ago
055-assign...	in a few seconds
056-data-di...	2 months ago
my_predicto...	2 months ago
my_predicto...	2 months ago

055-assignment.ipynb

054-gradient-boosting.ipynb

Python 3 (pykernel)

```

[20]: # Plot class balance
df["bankrupt"].value_counts(normalize=True).plot(
    kind = "bar",
    xlabel = "Bankrupt",
    ylabel = "Frequency",
    title = "Class Balance"
);
# Don't delete the code below
plt.savefig("Images/5-5-7.png", dpi=150)

```

```

[22]: with open("Images/5-5-7.png", "rb") as file:
      wqet_grader.grade("Project 5 Assessment", "Task 5.5.7", file)

```

Very impressive.  
Score: 1

Split

Simple 0 5 Python 3 (pykernel) | Idle

Mode: Command Ln 1, Col 1 English (United States) 055-assignment.ipynb

My Path Module | WorldsQuant U

work/ds-curricu (2) - JupyterLab

Data-Science-Lab/GridSearchCV

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name	Last Modified
050-bankruptcy-in-poland	2 days ago
051-workin...	2 months ago
052-imbalan...	4 days ago
053-random...	4 days ago
054-gradien...	2 months ago
055-assign...	in a few seconds
056-data-di...	2 months ago
my_predicto...	2 months ago
my_predicto...	2 months ago

055-assignment.ipynb

054-gradient-boosting.ipynb

Python 3 (pykernel)

Split

Task 5.5.8: Create your feature matrix  $X$  and target vector  $y$ . Your target is "bankrupt".

```

[23]: target = "bankrupt"
X = df.drop(columns="bankrupt")
y = df[target]

print("X shape:", X.shape)
print("y shape:", y.shape)

X shape: (6137, 95)
y shape: (6137,)

```

```

[24]: wqet_grader.grade("Project 5 Assessment", "Task 5.5.8a", X)

```

Correct.  
Score: 1

```

[27]: wqet_grader.grade("Project 5 Assessment", "Task 5.5.8b", y)

```

Good work!  
Score: 1

Task 5.5.9: Divide your dataset into training and test sets using a randomized split. Your test set should be 20% of your data. Be sure to set `random_state` to 42.

```

[28]: X_train, X_test, y_train, y_test = train_test_split(
      X, y, test_size=0.2, random_state=42
)

```

Simple 0 5 Python 3 (pykernel) | Idle

Mode: Command Ln 1, Col 1 English (United States) 055-assignment.ipynb

My Path Module | WorldsQuant | work/ds-curricu - JupyterLab | Data-Science-Lab/GridSearchCV | +

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

055-assignment.ipynb

054-gradient-boosting.ipynb

Python 3 (pykernel)

Filter files by name

055-assignment.ipynb

056-data-di...

057-predicto...

058-predicto...

055-assignment.ipynb

056-data-di...

057-predicto...

058-predicto...

Task 5.5.9: Divide your dataset into training and test sets using a randomized split. Your test set should be 20% of your data. Be sure to set `random_state` to 42.

```
[28]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

print("X_train shape:", X_train.shape)
print("y_train shape:", y_train.shape)
print("X_test shape:", X_test.shape)
print("y_test shape:", y_test.shape)

X_train shape: (4909, 95)
y_train shape: (4909,)
X_test shape: (1228, 95)
y_test shape: (1228,)
```

✓

Very impressive.  
Score: 1

Resample

Task 5.5.10: Create a new feature matrix `X_train_over` and target vector `y_train_over` by performing random over-sampling on the training data. Be sure to set the `random_state` to 42.

```
[31]: over_sampler = RandomOverSampler(random_state=42)
X_train_over, y_train_over = over_sampler.fit_resample(X_train, y_train)
print("X_train_over shape:", X_train_over.shape)
X_train_over.head()

X_train_over shape: (9512, 95)
```

	feat_1	feat_2	feat_3	feat_4	feat_5	feat_6	feat_7	feat_8	feat_9	feat_10	...	feat_86	feat_87	feat_88	feat_89	feat_90	feat_91	feat_92	feat_93	feat_94	feat_95
0	0.535855	0.599160	0.594411	0.627099	0.627099	0.999220	0.797686	0.809591	0.303518	0.781865	-	0.834091	0.022025	0.624364	0.627101	0.841977	0.275384	0.026791	0.565158	1	0.147943
1	0.554136	0.612734	0.595000	0.607388	0.607388	0.999120	0.797614	0.809483	0.303600	0.781754	-	0.840293	0.002407	0.624548	0.607385	0.842645	0.276532	0.026791	0.565158	1	0.062544
2	0.549554	0.603467	0.599122	0.620166	0.620166	0.999119	0.797569	0.809470	0.303524	0.781740	-	0.840403	0.000840	0.624010	0.620163	0.842873	0.277249	0.026800	0.565200	1	0.047929

✓

Boom! You got it.  
Score: 1

Simple

0

5

Python 3 (pykernel) | Idle

Mode Command

Ln 1, Col 1

English (United States)

055-assignment.ipynb

26°C

Haze

Search

ENG

23:37

17-03-2023

My Path Module | WorldsQuant | work/ds-curricu - JupyterLab | Data-Science-Lab/GridSearchCV | +

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

055-assignment.ipynb

054-gradient-boosting.ipynb

Python 3 (pykernel)

Filter files by name

055-assignment.ipynb

056-data-di...

057-predicto...

058-predicto...

055-assignment.ipynb

056-data-di...

057-predicto...

058-predicto...

Resample

Task 5.5.10: Create a new feature matrix `X_train_over` and target vector `y_train_over` by performing random over-sampling on the training data. Be sure to set the `random_state` to 42.

```
[31]: over_sampler = RandomOverSampler(random_state=42)
X_train_over, y_train_over = over_sampler.fit_resample(X_train, y_train)
print("X_train_over shape:", X_train_over.shape)
X_train_over.head()

X_train_over shape: (9512, 95)
```

	feat_1	feat_2	feat_3	feat_4	feat_5	feat_6	feat_7	feat_8	feat_9	feat_10	...	feat_86	feat_87	feat_88	feat_89	feat_90	feat_91	feat_92	feat_93	feat_94	feat_95
0	0.535855	0.599160	0.594411	0.627099	0.627099	0.999220	0.797686	0.809591	0.303518	0.781865	-	0.834091	0.022025	0.624364	0.627101	0.841977	0.275384	0.026791	0.565158	1	0.147943
1	0.554136	0.612734	0.595000	0.607388	0.607388	0.999120	0.797614	0.809483	0.303600	0.781754	-	0.840293	0.002407	0.624548	0.607385	0.842645	0.276532	0.026791	0.565158	1	0.062544
2	0.549554	0.603467	0.599122	0.620166	0.620166	0.999119	0.797569	0.809470	0.303524	0.781740	-	0.840403	0.000840	0.624010	0.620163	0.842873	0.277249	0.026800	0.565200	1	0.047929
3	0.543801	0.603249	0.606992	0.622515	0.622515	0.999259	0.797728	0.809649	0.303510	0.781930	-	0.831514	0.006176	0.626775	0.622513	0.842989	0.280013	0.026839	0.565375	1	0.028386
4	0.498659	0.562364	0.546978	0.603670	0.603670	0.998904	0.797584	0.809459	0.304000	0.781713	-	0.811988	0.004256	0.623674	0.603669	0.841105	0.277628	0.026897	0.565618	1	0.043080

✓

Boom! You got it.  
Score: 1

Build Model

Iterate

Simple

0

5

Python 3 (pykernel) | Idle

Mode Command

Ln 1, Col 1

English (United States)

055-assignment.ipynb

26°C

Haze

Search

ENG

23:37

17-03-2023

My Path Module | WorldsQuant U | work/ds-curricu (2) - JupyterLab | Data-Science-Lab/GridSearchCV | +

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name	Last Modified
.ipynb_chec...	2 days ago
data	2 months ago
images	4 days ago
051-workin...	4 days ago
052-imbalan...	2 months ago
053-random...	2 months ago
054-gradien...	an hour ago
055-assign...	seconds ago
056-data-di...	2 months ago
my_predicto...	2 months ago
my_predicto...	2 months ago

## Build Model

### Iterate

**Task 5.5.11:** Create a classifier `c1f` that can be trained on `(X_train_over, y_train_over)`. You can use any of the predictors you've learned about in the Data Science Lab.

```
[34]: c1f = GradientBoostingClassifier(random_state=42)
[34]: c1f
[34]: GradientBoostingClassifier
[34]: GradientBoostingClassifier(random_state=42)
[35]: wqet_grader.grade("Project 5 Assessment", "Task 5.5.11", c1f)
```

✓ Score: 1

**Task 5.5.12:** Perform cross-validation with your classifier using the over-sampled training data, and assign your results to `cv_scores`. Be sure to set the `cv` argument to 5.

**Tip:** Use your CV scores to evaluate different classifiers. Choose the one that gives you the best scores.

```
[36]: cv_scores = cross_val_score(c1f, X_train_over, y_train_over, cv=5, n_jobs=-1)
[36]: print(cv_scores)
[36]: [0.96952181 0.97162375 0.97003155 0.97160883 0.96845426]
[37]: cv_scores
[37]: array([0.96952181, 0.97162375, 0.97003155, 0.97160883, 0.96845426])
```

Simple 0 5 Python 3 (pykernel) | Idle Mode: Command Ln 1, Col 1 English (United States) 055-assignment.ipynb

26°C Haze

My Path Module | WorldsQuant U | work/ds-curricu (2) - JupyterLab | Data-Science-Lab/GridSearchCV | +

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name	Last Modified
.ipynb_chec...	2 days ago
data	2 months ago
images	4 days ago
051-workin...	4 days ago
052-imbalan...	2 months ago
053-random...	2 months ago
054-gradien...	an hour ago
055-assign...	seconds ago
056-data-di...	2 months ago
my_predicto...	2 months ago
my_predicto...	2 months ago

**Task 5.5.12:** Perform cross-validation with your classifier using the over-sampled training data, and assign your results to `cv_scores`. Be sure to set the `cv` argument to 5.

**Tip:** Use your CV scores to evaluate different classifiers. Choose the one that gives you the best scores.

```
[36]: cv_scores = cross_val_score(c1f, X_train_over, y_train_over, cv=5, n_jobs=-1)
[36]: print(cv_scores)
[36]: [0.96952181 0.97162375 0.97003155 0.97160883 0.96845426]
[37]: cv_scores
[37]: array([0.96952181, 0.97162375, 0.97003155, 0.97160883, 0.96845426])
[38]: wqet_grader.grade("Project 5 Assessment", "Task 5.5.12", list(cv_scores))
```

✓ Yes! Great problem solving. Score: 1

**Ungraded Task:** Create a dictionary `params` with the range of hyperparameters that you want to evaluate for your classifier. If you're not sure which hyperparameters to tune, check the [scikit-learn](#) documentation for your predictor for ideas.

**Tip:** If the classifier you built is a predictor only (not a pipeline with multiple steps), you don't need to include the step name in the keys of your `params` dictionary. For example, if your classifier was only a random forest (not a pipeline containing a random forest), you would access the number of estimators using `"n_estimators"`, not `"randomforestclassifier__n_estimators"`.

```
[39]: params = params = {
[39]: "n_estimators": range(20, 31, 5),
[39]: "max_depth": range(2, 5)
[39]: }
[39]: params
[39]: {'n_estimators': range(20, 31, 5), 'max_depth': range(2, 5)}
```

**Task 5.5.13:** Create a `GridSearchCV` named `model` that includes your classifier and hyperparameter grid. Be sure to set `cv` to 5, `n_jobs` to -1, and `verbose` to 1.

Simple 0 5 Python 3 (pykernel) | Idle Mode: Command Ln 1, Col 1 English (United States) 055-assignment.ipynb

26°C Haze



My Path Module | WorldsQuant | work/ds-curricu (2) - JupyterLab | Data-Science-Lab/GridSearchCV | +

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

055-assignment.ipynb 054-gradient-boosting.ipynb

Python 3 (pykernel)

Filter files by name

Name Last Modified

- 055-assignment.ipynb seconds ago
- 056-data-di... 2 months ago
- my\_predicto... 2 months ago
- my\_predicto... 2 months ago

Task 5.5.13: Create a `GridSearchCV` named `model` that includes your classifier and hyperparameter grid. Be sure to set `cv` to 5, `n_jobs` to -1, and `verbose` to 1.

```
[40]: model = GridSearchCV(
      |     clf,
      |     param_grid=params,
      |     cv=5,
      |     n_jobs=-1,
      |     verbose=1
      | )
```

```
[41]: wqet_grader.grade("Project 5 Assessment", "Task 5.5.13", model)
```

Booni! You got it.  
Score: 1

Ungraded Task: Fit your model to the over-sampled training data.

```
[46]: model.fit(X_train_over, y_train_over)
```

Fitting 5 folds for each of 9 candidates, totalling 45 fits

```
[46]: *
      | GridSearchCV
      | * estimator: GradientBoostingClassifier
      |   * GradientBoostingClassifier
```

Task 5.5.14: Extract the cross-validation results from your model, and load them into a DataFrame named `cv_results`. Looking at the results, which set of hyperparameters led to the best performance?

```
[48]: cv_results = pd.DataFrame(model.cv_results_)
      | cv_results.head(5)
```

Simple 0 5 Python 3 (pykernel) | Idle

26°C Haze

My Path Module | WorldsQuant | work/ds-curricu (2) - JupyterLab | Data-Science-Lab/GridSearchCV | +

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

055-assignment.ipynb 054-gradient-boosting.ipynb

Python 3 (pykernel)

Filter files by name

Name Last Modified

- 055-assignment.ipynb seconds ago
- 056-data-di... 2 months ago
- my\_predicto... 2 months ago
- my\_predicto... 2 months ago

Task 5.5.14: Extract the cross-validation results from your model, and load them into a DataFrame named `cv_results`. Looking at the results, which set of hyperparameters led to the best performance?

```
[48]: cv_results = pd.DataFrame(model.cv_results_)
      | cv_results.head(5)
```

	mean_fit_time	std_fit_time	mean_score_time	std_score_time	param_max_depth	param_n_estimators	params	split0_test_score	split1_test_score	split2_test_score	split3_test_score	split4_test_score	mean_test_score
0	4.274908	0.119663	0.004303	0.000142	2	20	{'max_depth': 2, 'n_estimators': 20}	0.909616	0.897530	0.903260	0.905363	0.906414	0.904457
1	5.252297	0.130287	0.028929	0.029672	2	25	{'max_depth': 2, 'n_estimators': 25}	0.912769	0.913820	0.917455	0.913775	0.912198	0.913820
2	6.512919	0.120718	0.005092	0.001144	2	30	{'max_depth': 2, 'n_estimators': 30}	0.923279	0.917499	0.916930	0.923239	0.919558	0.919558
3	6.279949	0.180477	0.017523	0.024972	3	20	{'max_depth': 3, 'n_estimators': 20}	0.929585	0.930636	0.932177	0.934805	0.931651	0.931651
4	7.655999	0.134539	0.004829	0.000464	3	25	{'max_depth': 3, 'n_estimators': 25}	0.935365	0.931687	0.939537	0.937434	0.935331	0.935331

```
[49]: wqet_grader.grade("Project 5 Assessment", "Task 5.5.14", cv_results)
```

Yes! Great problem solving.  
Score: 1

My Path Module | WorldsQuant U

work/ds-curricu (2) - JupyterLab

Data-Science-Lab/conf\_matrix.py

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

055-assignment.ipynb

054-gradient-boosting.ipynb

Python 3 (pykernel)

Python 3 (pykernel)

Filter files by name

ds-curriculum / 050-bankruptcy-in-poland /

Name	Last Modified
.ipynb_chec...	2 days ago
data	2 months ago
images	2 minutes ago
051-workin...	4 days ago
052-imbalan...	2 months ago
053-random...	2 months ago
054-gradien...	an hour ago
055-assign...	2 minutes ago
056-data-di...	2 months ago
my_predicto...	2 months ago
my_predicto...	2 months ago

Task 5.5.15: Extract the best hyperparameters from your model and assign them to `best_params`.

```
[51]: best_params = model.best_params_
      print(best_params)
      {'max_depth': 4, 'n_estimators': 30}

[52]: wqet_grader.grade(
      "Project 5 Assessment", "Task 5.5.15", [isinstance(best_params, dict)])
      ]
```

✓

You are coding

Score: 1

Evaluate

Ungraded Task: Test the quality of your model by calculating accuracy scores for the training and test data.

```
[53]: acc_train = model.score(X_train, y_train)
      acc_test = model.score(X_test, y_test)

      print("Model Training Accuracy:", round(acc_train, 4))
      print("Model Test Accuracy:", round(acc_test, 4))

      Model Training Accuracy: 0.9466
      Model Test Accuracy: 0.9389

Task 5.5.16: Plot a confusion matrix that shows how your model performed on your test set.
```

```
[59]: ConfusionMatrixDisplay.from_estimator(model, X_test, y_test);
      # Don't delete the code below
      plt.savefig("images/5-5-16.png", dpi=150)
```

Simple

0 5

Python 3 (pykernel) | Idle

Mode: Command

Ln 1, Col 1

English (United States)

055-assignment.ipynb

My Path Module | WorldsQuant U

work/ds-curricu (2) - JupyterLab

Data-Science-Lab/conf\_matrix.py

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

055-assignment.ipynb

054-gradient-boosting.ipynb

Python 3 (pykernel)

Python 3 (pykernel)

Filter files by name

ds-curriculum / 050-bankruptcy-in-poland /

Name	Last Modified
.ipynb_chec...	2 days ago
data	2 months ago
images	3 minutes ago
051-workin...	4 days ago
052-imbalan...	2 months ago
053-random...	2 months ago
054-gradien...	an hour ago
055-assign...	2 minutes ago
056-data-di...	2 months ago
my_predicto...	2 months ago
my_predicto...	2 months ago

Task 5.5.16: Plot a confusion matrix that shows how your model performed on your test set.

```
[59]: ConfusionMatrixDisplay.from_estimator(model, X_test, y_test);
      # Don't delete the code below
      plt.savefig("images/5-5-16.png", dpi=150)
```

	True	False
True	30	7
False	68	1123

✓

Way to go!

Score: 1

Task 5.5.17: Generate a classification report for your model's performance on the test data and assign it to `class_report`.

```
[ ]: class_report = classification_report(y_test, model.predict(X_test))
      print(class_report)

[ ]: wqet_grader.grade("Project 5 Assessment", "Task 5.5.17", class_report)
```

Simple

0 5

Python 3 (pykernel) | Idle

Mode: Command

Ln 1, Col 1

English (United States)

055-assignment.ipynb



My Path Module | WorldsQuant U... work/ds-curricu (2) - JupyterLab Data-Science-Lab/conf\_matrix.py

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- 055-assignment.ipynb in a few seconds
- 056-data-di...
- my\_predicto...
- my\_predicto...

055-assignment.ipynb

```
[60]: with open("images/5-5-16.png", "rb") as file:
      wqet_grader.grade("Project 5 Assessment", "Task 5.5.16", file)
```

Way to go!  
Score: 1

**Task 5.5.17:** Generate a classification report for your model's performance on the test data and assign it to `class_report`.

```
[61]: class_report = classification_report(y_test, model.predict(X_test))
      print(class_report)
```

	precision	recall	f1-score	support
False	0.99	0.94	0.97	1191
True	0.31	0.81	0.44	37
accuracy			0.94	1228
macro avg	0.65	0.88	0.71	1228
weighted avg	0.97	0.94	0.95	1228

```
[62]: wqet_grader.grade("Project 5 Assessment", "Task 5.5.17", class_report)
```

You = coding  
Score: 1

## Communicate

**Task 5.5.18:** Create a horizontal bar chart with the 10 most important features for your model. Be sure to label the x-axis "Gini Importance", the y-axis "Feature", and use the title "Feature Importance".

26°C Haze

My Path Module | WorldsQuant U... work/ds-curricu (2) - JupyterLab Data-Science-Lab/barh.py at ma... |

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- 055-assignment.ipynb a minute ago
- 056-data-di...
- my\_predicto...
- my\_predicto...

055-assignment.ipynb

## Communicate

**Task 5.5.18:** Create a horizontal bar chart with the 10 most important features for your model. Be sure to label the x-axis "Gini Importance", the y-axis "Feature", and use the title "Feature Importance".

```
[67]: features = X_train_over.columns

# Extract importances from model
importances = model.best_estimator_.feature_importances_

# Create a series with feature names and importances
feat_imp = pd.Series(importances, index=features).sort_values()

# Plot 10 most important features
feat_imp.tail(10).plot(kind="barh")
plt.xlabel("Gini Importance")
plt.ylabel("Feature")
plt.title("Feature Importance");
# Don't delete the code below
plt.savefig("images/5-5-17.png", dpi=150)
```

Feature Importance

feat\_86  
feat\_19  
feat\_36  
feat\_40  
feat\_9  
feat\_46  
feat\_69  
feat\_16  
feat\_3  
feat\_34

Gini Importance

My Path Module | WorldQuant U x work/ds-curricu (2) - JupyterLab x Data-Science-Lab/050-bankrupty-in-poland/055-assignment.ipynb

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name	Last Modified
.ipynb_chec...	2 days ago
data	2 months ago
images	4 minutes ago
051-workin...	4 days ago
052-imbalan...	2 months ago
053-random...	2 months ago
054-gradien...	2 hours ago
055-assign...	a minute ago
056-data-di...	2 months ago
model-5-5.pkl	seconds ago
my_predicto...	2 months ago
my_predicto...	2 months ago

055-assignment.ipynb

054-gradient-boosting.ipynb x

055-assignment.ipynb

Markdown

Python 3 (pykernel)

wget\_grader.grade("Project 5 Assessment", "Task 5.5.18", file)

Yes! Your hard work is paying off.  
Score: 1

**Task 5.5.19:** Save your best-performing model to a file named "model-5-5.pkl".

```
[69]: # Save model
with open("model-5-5.pkl", "wb") as f:
    pickle.dump(model, f)

# Load model from "Destination"
with open("model-5-5.pkl", "rb") as f:
    loaded_model = pickle.load(f)
print(loaded_model)

GridSearchCV(cv=5, estimator=GradientBoostingClassifier(random_state=42),
              n_jobs=-1,
              param_grid={'max_depth': range(2, 5),
                           'n_estimators': range(20, 31, 5)},
              verbose=1)
```

[70]: with open("model-5-5.pkl", "rb") as f:
 wget\_grader.grade("Project 5 Assessment", "Task 5.5.19", pickle.load(f))

Yes! Keep on rockin'. That's right.  
Score: 1

**Task 5.5.20:** Open the file my\_predictor\_assignment.py. Add your wrangle function, and then create a make\_predictions function that takes two arguments: data\_filepath and model\_filepath. Use the cell below to test your module. When you're satisfied with the result, submit it to the grader.

```
[72]: # Import your module
```

Simple 0 5 Python 3 (pykernel) | Idle

Mode: Command Ln 1, Col 1 English (United States) 055-assignment.ipynb

26°C  
Haze

Applied Data Science Lab x My Path Module | WorldQuant U x work/ds-curricu (3) - JupyterLab x (45) YouTube x

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name	Last Modified
poland-ban...	2 months ago
poland-ban...	2 months ago
taiwan-ban...	2 months ago
taiwan-ban...	2 months ago

055-assignment.ipynb

my\_predictor\_assignment.py x

054-gradient-boosting.ipynb x

Code

Python 3 (pykernel)

**Task 5.5.20:** Open the file my\_predictor\_assignment.py. Add your wrangle function, and then create a make\_predictions function that takes two arguments: data\_filepath and model\_filepath. Use the cell below to test your module. When you're satisfied with the result, submit it to the grader.

```
[62]: %bash
cat my_predictor_assignment.py

# Import libraries
import gzip
import json
import pickle
import pandas as pd

# Add wrangle function from lesson 5.4
def wrangle(filePath):
    # Open compressed file, load to dict
    with gzip.open(filePath, "r") as f:
        data = json.load(f)

    # Dictionary --> DataFrame, set index
    df = pd.DataFrame().from_dict(data["observations"]).set_index("id")

    return df
df = wrangle('data/taiwan-bankruptcy-data-test-features.json.gz')
print("df shape:", df.shape)
df.head()

# Add make_predictions function from lesson 5.3
def make_predictions(data_filepath, model_filepath):
    X_test = wrangle(data_filepath)
    with open(model_filepath, "rb") as f:
        model = pickle.load(f)
    y_test_pred = model.predict(X_test)
    y_test_pred = pd.Series(y_test_pred, index=X_test.index, name="bankrupt")
    return y_test_pred

[60]: from my_predictor_assignment import make_predictions
```

Simple 0 6 Python 3 (pykernel) | Idle

Mode: Command Ln 1, Col 1 English (United States) 055-assignment.ipynb

34°C  
Haze

VARIABLES

CALLSTACK

BREAKPOINTS

SOURCE

17-03-2023 23:48

18-03-2023 14:57

Applied Data Science Lab | My Path Module | WorldQuant | work/ds-curriculum (3) - JupyterLab | (45) YouTube

vm.wqu.edu/lab/tree/work/ds-curriculum/050-bankruptcy-in-poland/055-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

055-assignment.ipynb | E: my\_predictor\_assignment.py | 054-gradient-boosting.ipynb | Python 3 (ipykernel)

Filter files by name

Name	Last Modified
/ / 050-bankruptcy-in-poland / data /	
poland-ban...	2 months ago
poland-ban...	2 months ago
taiwan-bank...	2 months ago
taiwan-bank...	2 months ago

```
[60]: from my_predictor_assignment import make_predictions

y_test_pred = make_predictions(
    data_filepath="data/taiwan-bankruptcy-data-test-features.json.gz",
    model_filepath="model-5-5.pkl",
)

print("predictions shape:", y_test_pred.shape)
y_test_pred.head()

df shape: (682, 95)
predictions shape: (682,)
```

[60]: id  
18 False  
20 False  
24 True  
32 True  
38 False  
Name: bankrupt, dtype: bool

Tip: If you get an **ImportError** when you try to import `make_predictions` from `my_predictor_assignment`, try restarting your kernel. Go to the **Kernel** menu and click on **Restart Kernel and Clear All Outputs**. Then rerun just the cell above. 🌟

```
[61]: wqet_grader.grade(
    "Project 5 Assessment",
    "Task 5.5.20",
    make_predictions(
        data_filepath="data/taiwan-bankruptcy-data-test-features.json.gz",
        model_filepath="model-5-5.pkl",
    ),
)
```

✓ Your model's accuracy score is 0.9179. Party time! 🎉  
Score: 1

Simple 0 6 Python 3 (ipykernel) | Idle Mode: Command Ln 1, Col 1 English (United States) 055-assignment.ipynb

34°C Haze Search 18-03-2023 14:57