INTRODUCTION TO PREDICTIVE ANALYTICS IN PYTHON

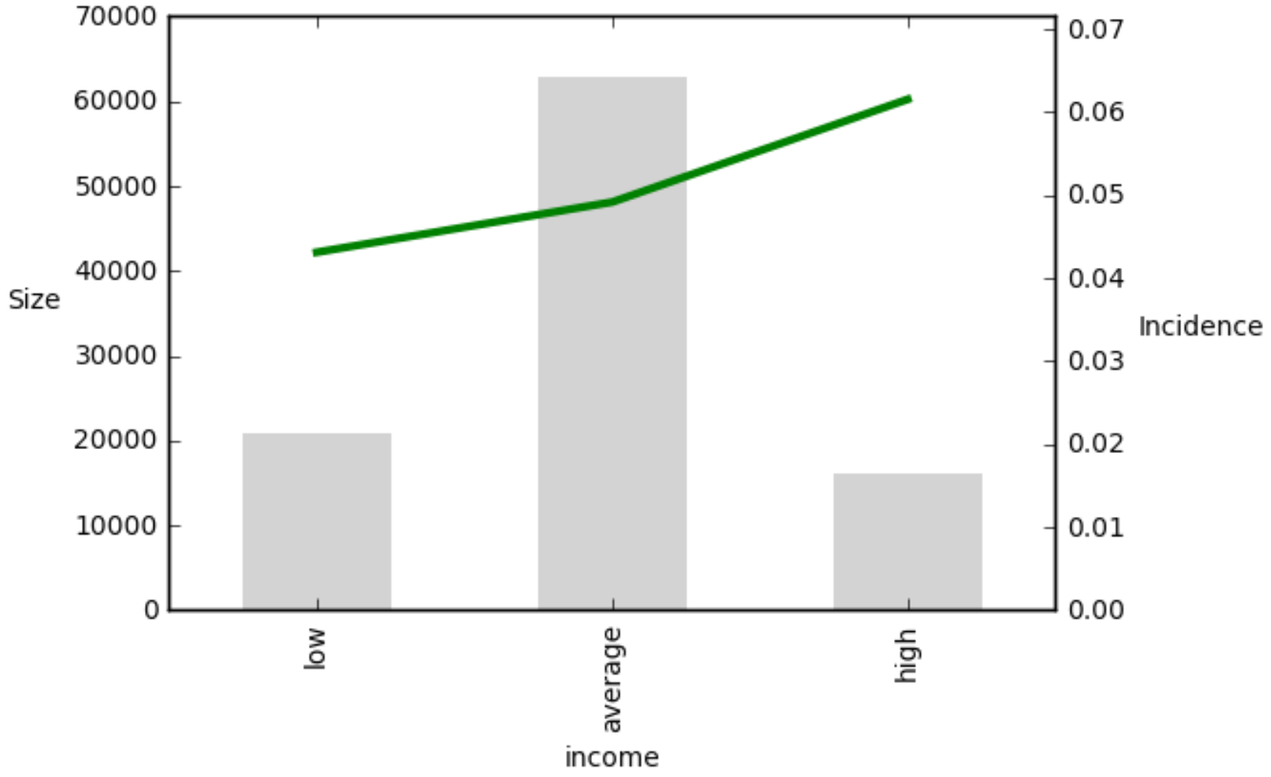
Predictor insight graphs

Nele Verbiest, Ph.D
Data Scientist
Python Predictions

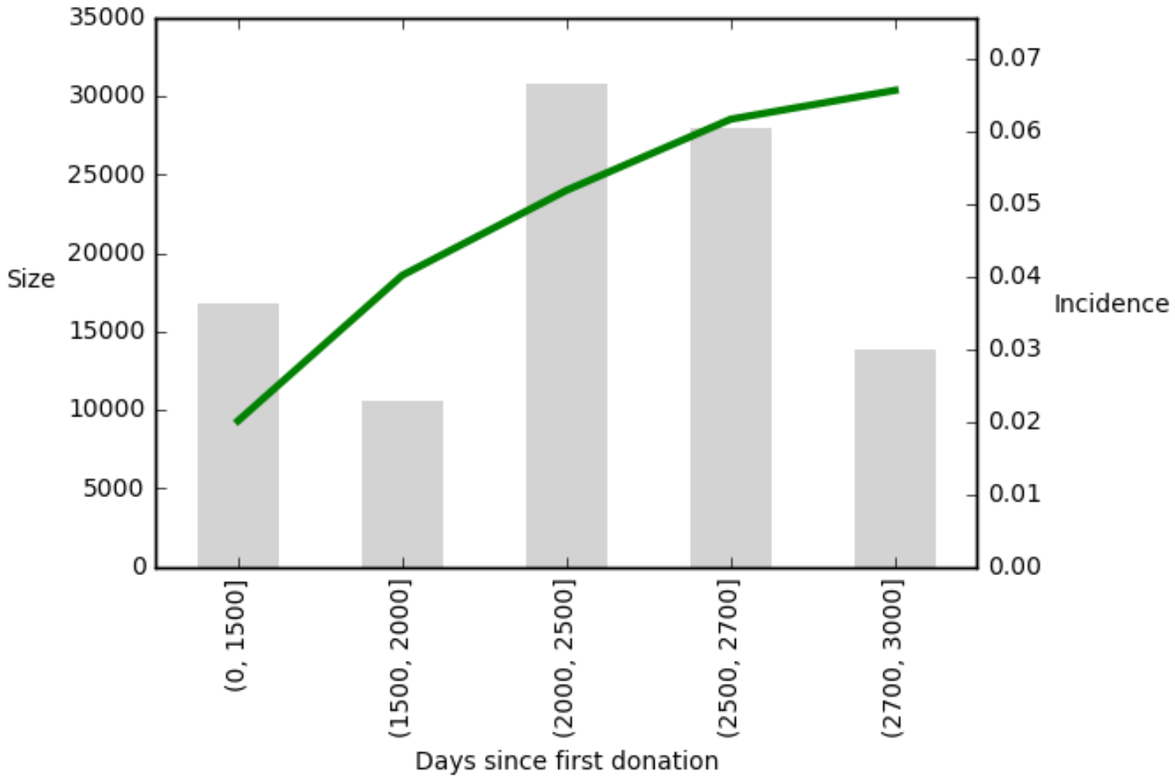
Motivation for predictor insight graphs

1. Build model
2. Evaluate model using AUC
3. Evaluate model using cumulative gains and lift curves
4. Verify whether the variables in the model are interpretable

Interpretation of predictor insight graphs



Predictor insight graphs for continuous variables



The predictor insight graph table

Income	Size	Incidence
low	20850	0.0431
average	62950	0.0492
high	16200	0.0615

```
print(pig_table["Size"][income=="low"])
```

```
20850
```

Constructing a predictor insight graph

- (Discretisation of variable if continuous)
- Calculate predictor insight graph table
- Plot the predictor insight graph

INTRODUCTION TO PREDICTIVE ANALYTICS IN PYTHON

Let's practice!

INTRODUCTION TO PREDICTIVE ANALYTICS IN PYTHON

Discretization of continuous variables

Nele Verbiest, Ph.D
Data Scientist
Python Predictions

Discretization in python

```
variable = "max_gift"
number_bins = 3
basetable["disc_max_gift"] = pd.qcut(basetable[variable], number_bins)

basetable.groupby("disc_max_gift").size()

disc_mean_gift
[2, 84.25]          33330
(84.25, 96.833]     33330
(96.833, 197]       33330
dtype: int64
```

Which variables should be discretized

```
variables_model = ["income_average","mean_gift","gender_M","min_gift","age"]
def check_discretize(basetable, variable, threshold):
    return(len(basetable.groupby(variable))>threshold)

check_discretize(basetable, "mean_gift",5)

True

check_discretize(basetable, "income_average",5)

False
```

Discretization of all variables

```
variables_model = ["income_average","mean_gift","gender_M","min_gift","age"]
def check_discretize(basetable, variable, threshold):
    return(len(basetable.groupby(variable))>threshold)

threshold = 5
number_bins = 5
for variable in variables_model:
    if check_discretize(basetable, variable, threshold):
        new_variable = "disc" + variable
        basetable[new_variable] = pd.qcut(basetable[variable], number_bins)
```

Clean cuts

```
basetable["disc_age"] = pd.qcut(basetable["age"], 5)
basetable["disc_age"].unique()

[(38, 49], (68, 110], [19, 38], (49, 59], (59, 68]]

basetable["disc_age"] = pd.cut(basetable["age"],[18,30,40,50,60,110])
basetable.groupby("disc_age").size()

disc_age
(18, 30]     10017
(30, 40]     14448
(40, 50]     19002
(50, 60]     24684
(60, 110]    31849
```

INTRODUCTION TO PREDICTIVE ANALYTICS IN PYTHON

Let's practice!

INTRODUCTION TO PREDICTIVE ANALYTICS IN PYTHON

Preparing the predictor insight graph table

Nele Verbiest, Ph.D
Data Scientist
Python Predictions

The predictor insight graph table

disc_mean_gift	Incidence	Size
[2, 78]	0.013042	20013
(78, 87]	0.029554	19997
(87, 94]	0.040831	20034
(94, 103]	0.063563	20405
(103, 197]	0.103524	19551

Calculating the predictor insight graph table

```
# Load the numpy module
import numpy as np
# Function that calculates the predictor insight graph table
def create_pig_table(df, target, variable):

    # Group by the variable you want to plot
    groups = df[[target,variable]].groupby(variable)

    # Calculate the size and incidence of each group
    pig_table = groups[target].agg({'Incidence' : np.mean, \
    'Size' : np.size}).reset_index()
    return pig_table

print(create_pig_table(basetable,"target","country")

country    Incidence    Size
India    0.050934    49849
UK    0.050512    10057
USA    0.048486    40094
```

Calculating multiple predictor insight graph tables

```
# Variables you like to plot.
variables = ["country", "gender", "disc_mean_gift", "age"]

# Empty dictionary.
pig_tables = {}

# Loop over all variables
for variable in variables:

    # Create the predictor insight graph table
    pig_table = create_pig_table(basetable, "target", variable)

    # Store the table in the dictionary
    pig_tables[variable] = pig_table

print(create_pig_table(basetable,"target","country")

country    Incidence    Size
India    0.050934    49849
UK    0.050512    10057
USA    0.048486    40094
```

INTRODUCTION TO PREDICTIVE ANALYTICS IN PYTHON

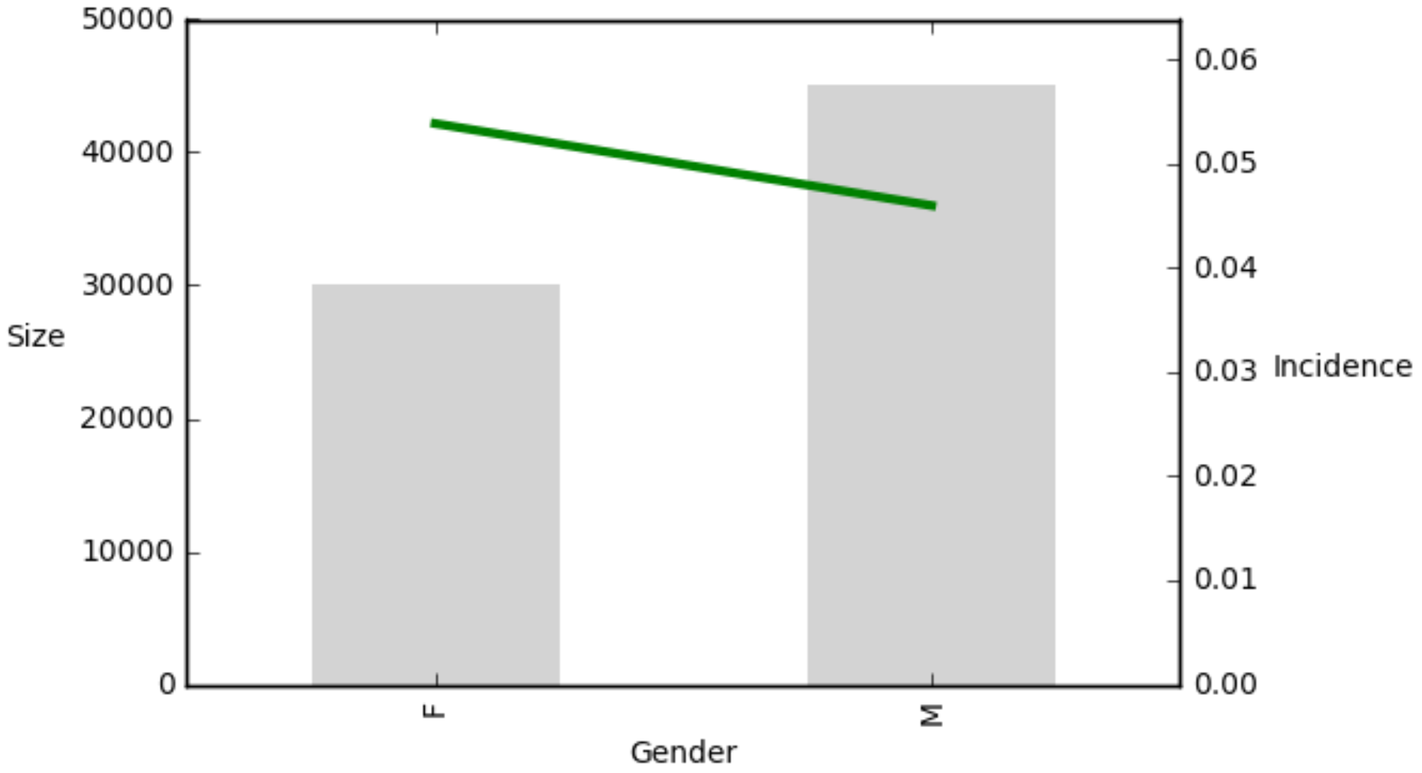
Let's practice!

INTRODUCTION TO PREDICTIVE ANALYTICS IN PYTHON

Plotting the predictor insight graph

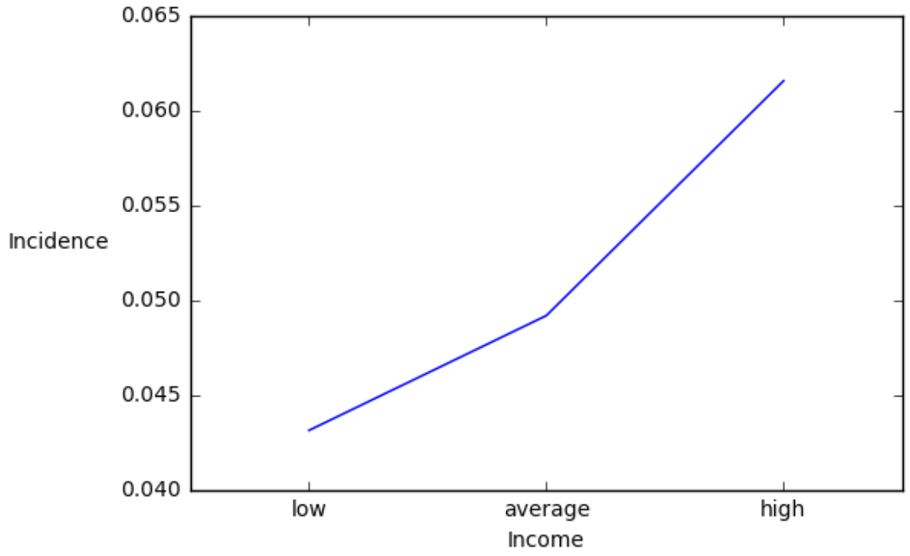
Nele Verbiest, Ph.D
Data Scientist
Python Predictions

The predictor insight graph



Plotting the target incidence

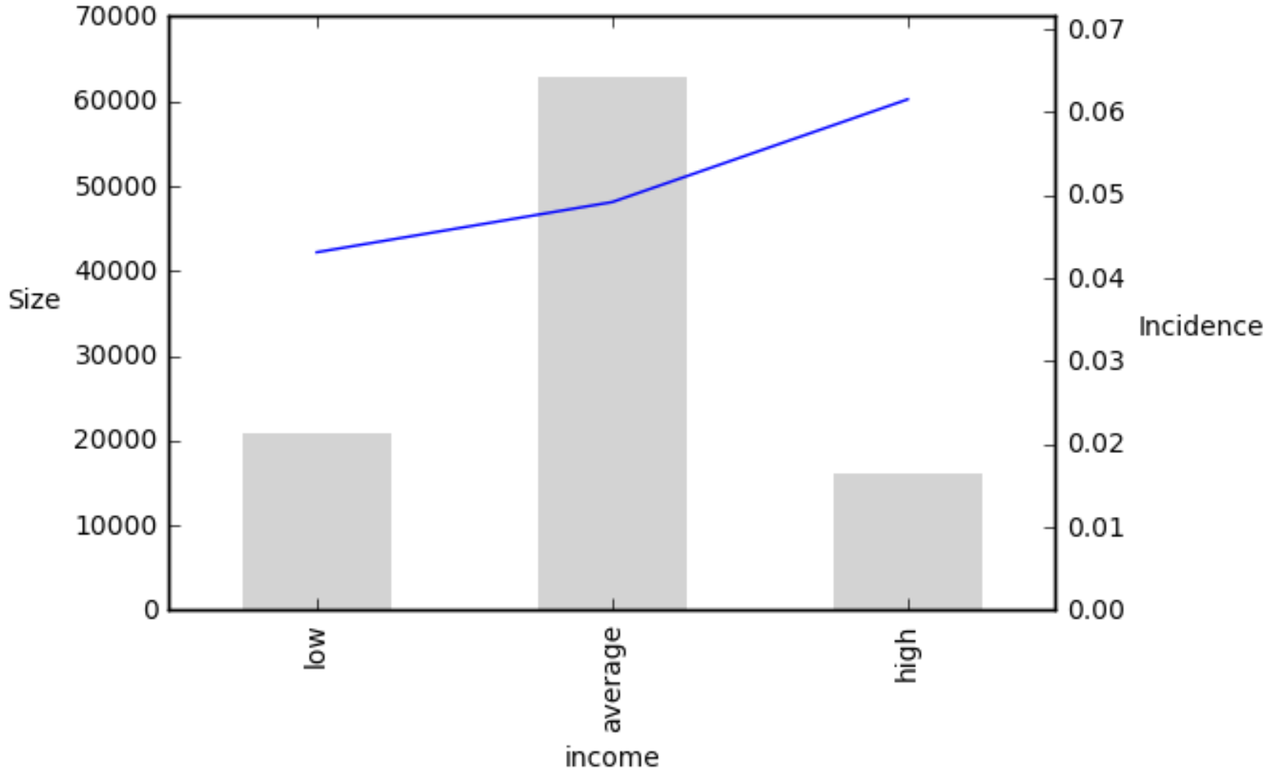
```
import matplotlib.pyplot as plt
import numpy as np

# Plot the graph
pig_table["Incidence"].plot()

# Show the group names
plt.xticks(np.arange(len(pig_table)),
    pig_table["income"])

# Center the groups names
width = 0.5
plt.xlim([-width, len(pig_table)-width

plt.ylabel("Incidence", rotation = 0,
    rotation_mode="anchor",
    ha = "right")
plt.xlabel("Income")
plt.show()
```



Plotting the sizes

```
import matplotlib.pyplot as plt
import numpy as np
# Plot the graph

plt.ylabel("Size", rotation = 0,rotation_mode="anchor", ha = "right" )

pig_table["Incidence"].plot(secondary_y = True)

pig_table["Size"].plot(kind='bar', width = 0.5,
                color = "lightgray", edgecolor = "none") ## Add bars

# Show the group names
plt.xticks(np.arange(len(pig_table)), pig_table["income"])
# Center the groups names
plt.xlim([-0.5, len(pt)-0.5])
plt.ylabel("Incidence", rotation = 0, rotation_mode="anchor", ha = "right"
plt.xlabel("Income")
plt.show()
```

Plotting the sizes



INTRODUCTION TO PREDICTIVE ANALYTICS IN PYTHON

Let's practice!

INTRODUCTION TO PREDICTIVE ANALYTICS IN PYTHON

Summary

Nele Verbiest
Data Scientist@Python Predictions

What you learned ... and what's up next?

1. Construct the basetable
2. Construct predictive models using logistic regression
3. Forward variable selection
4. Evaluation curves
5. Predictor insight graphs

INTRODUCTION TO PREDICTIVE ANALYTICS IN PYTHON

See you in the next course!