



## FOUNDATIONS OF PREDICTIVE ANALYTICS IN PYTHON (PART 1)

# The cumulative gains curve

Nele Verbiest, Ph.D

Data Scientist  
Python Predictions



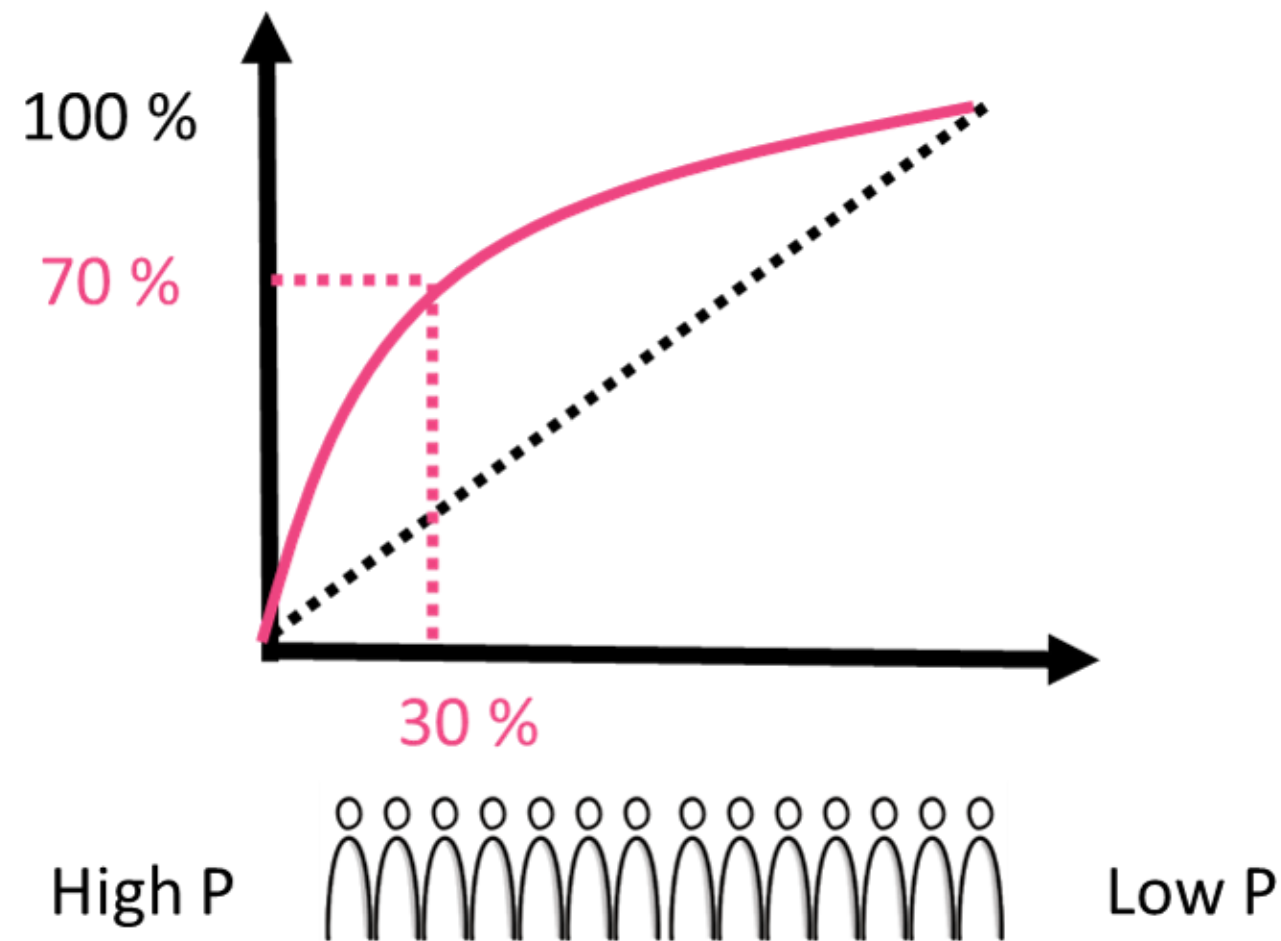
# Evaluation curves

AUC:

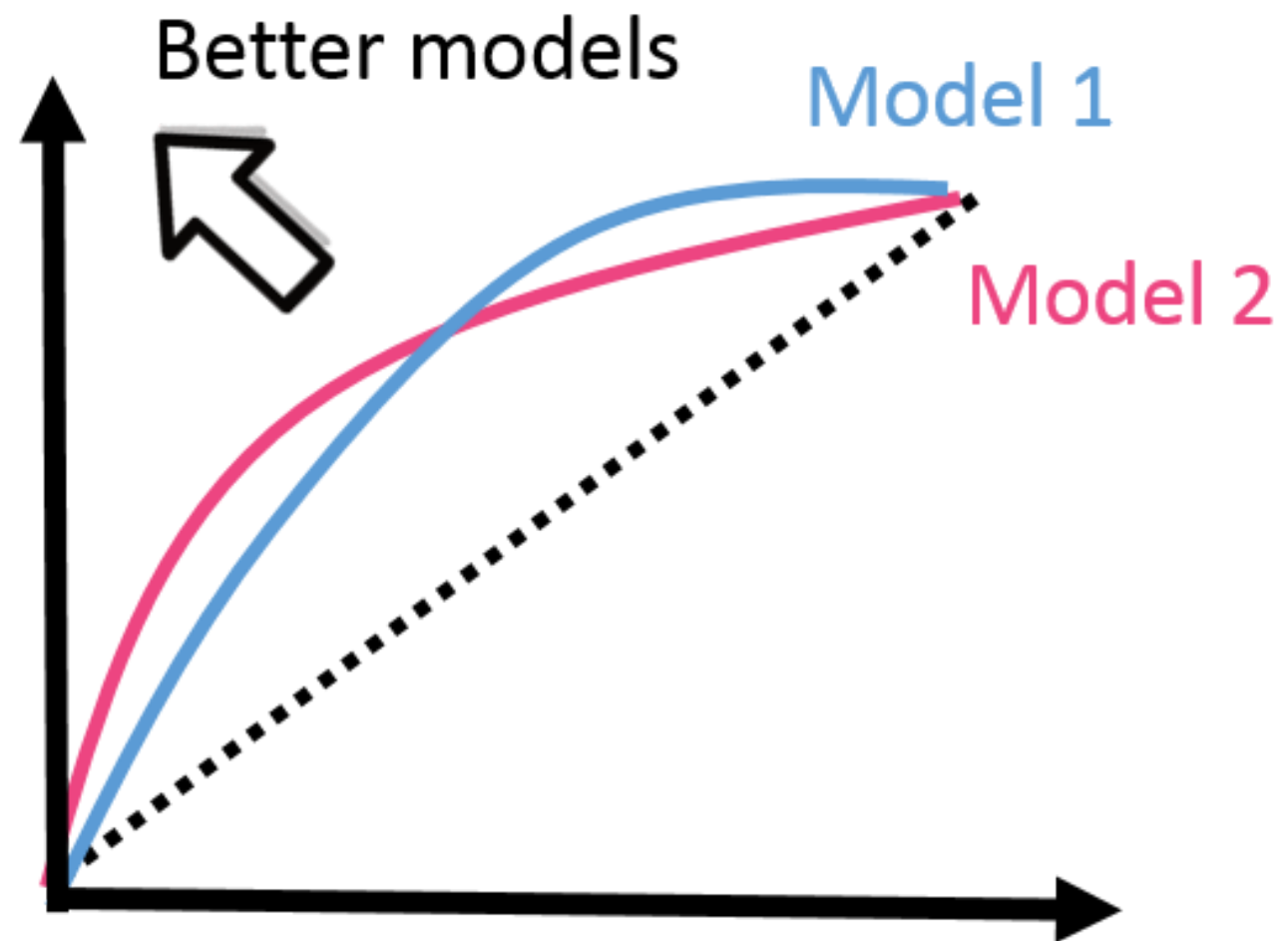
- Complex
- Single number



# Cumulative gains construction



# Cumulative gains interpretation





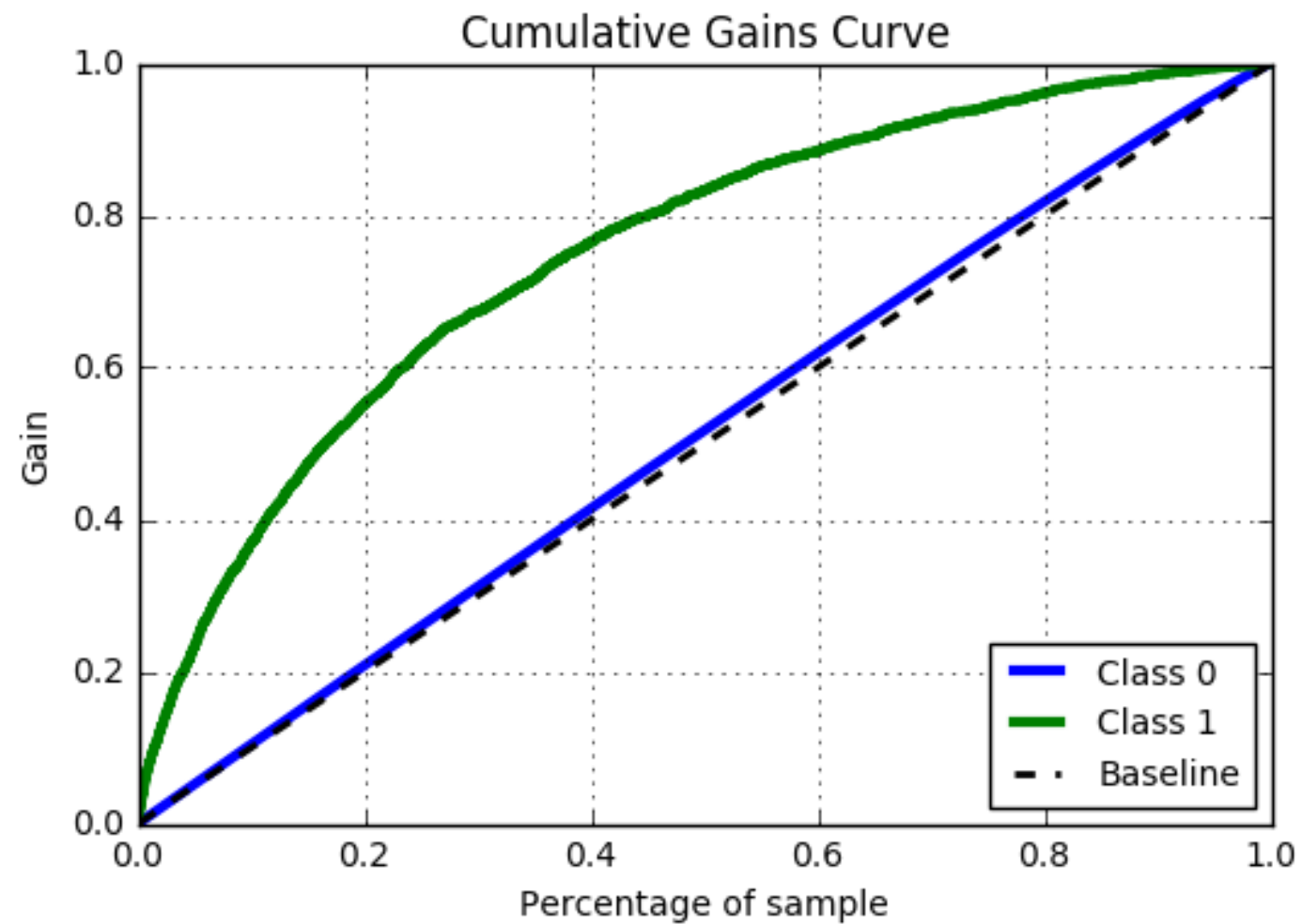
# Cumulative gains in Python

```
import scikitplot as skplt
import matplotlib.pyplot as plt

skplt.metrics.plot_cumulative_gain(true_values, predictions)
plt.show()
```



# Cumulative gains in Python





## FOUNDATIONS OF PREDICTIVE ANALYTICS IN PYTHON (PART 1)

**Let's practice!**



## FOUNDATIONS OF PREDICTIVE ANALYTICS IN PYTHON (PART 1)

# The lift curve

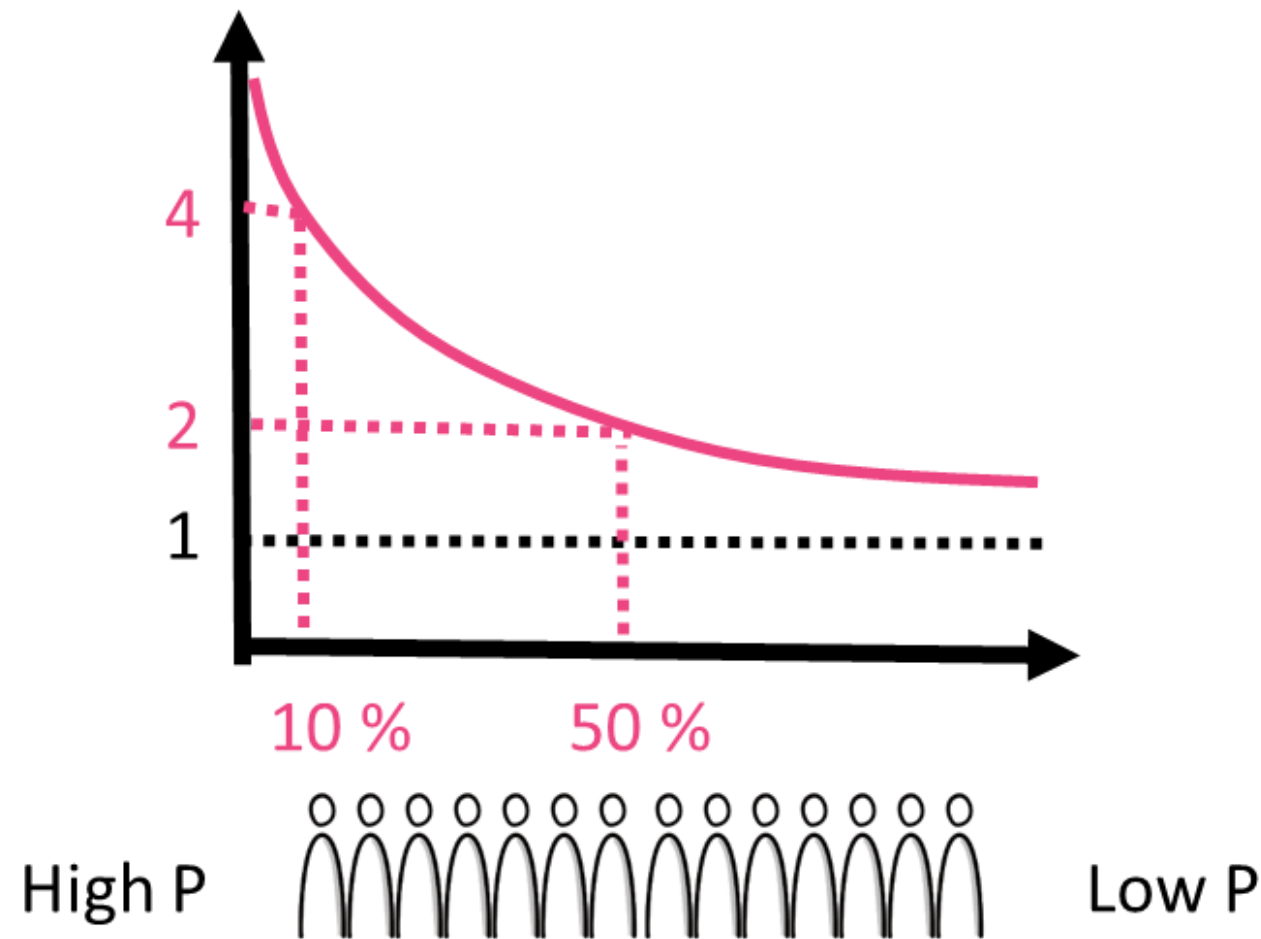
Nele Verbiest, Ph.D

Data Scientist  
Python Predictions

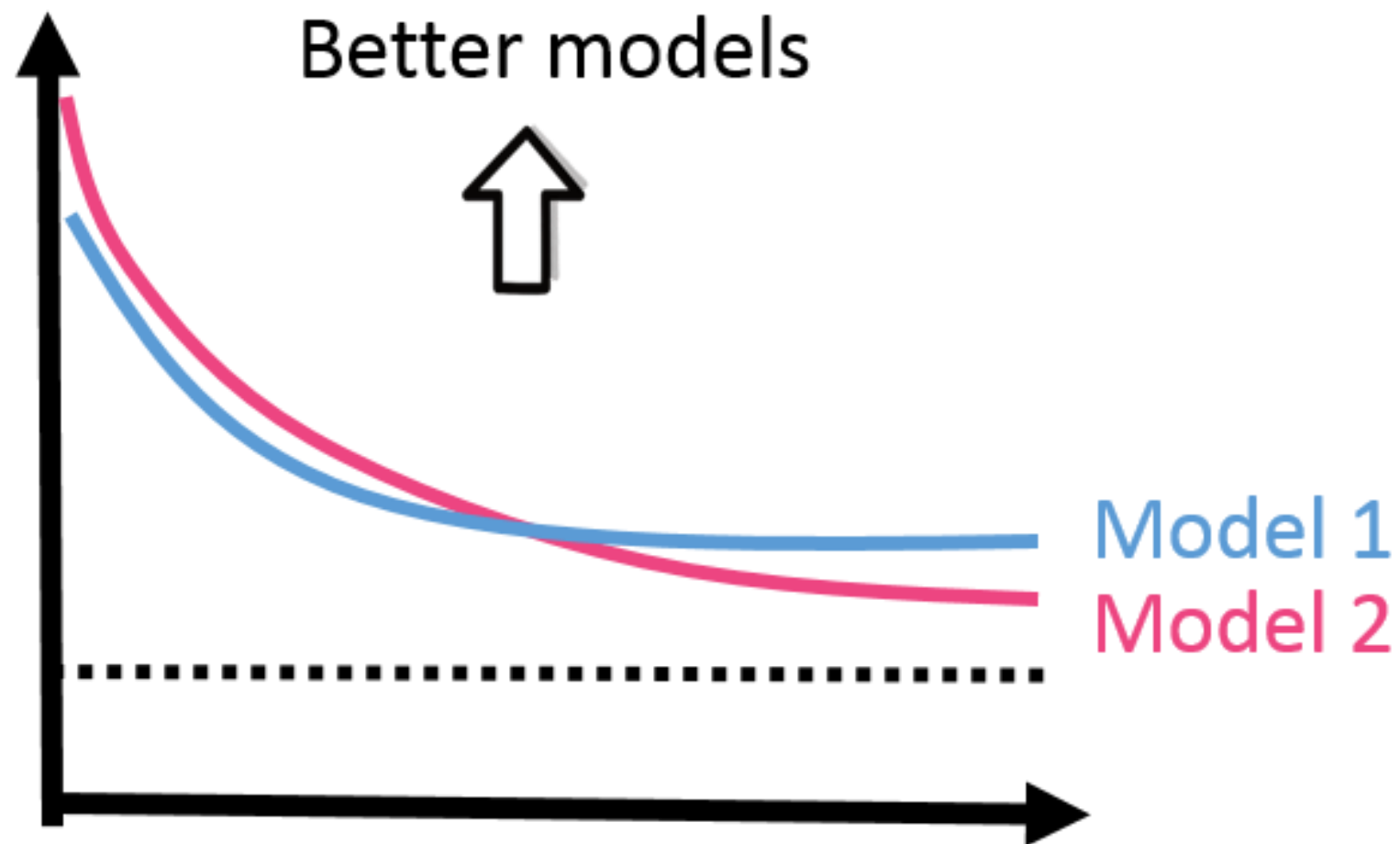




# Lift curve construction



# Lift curve interpretation





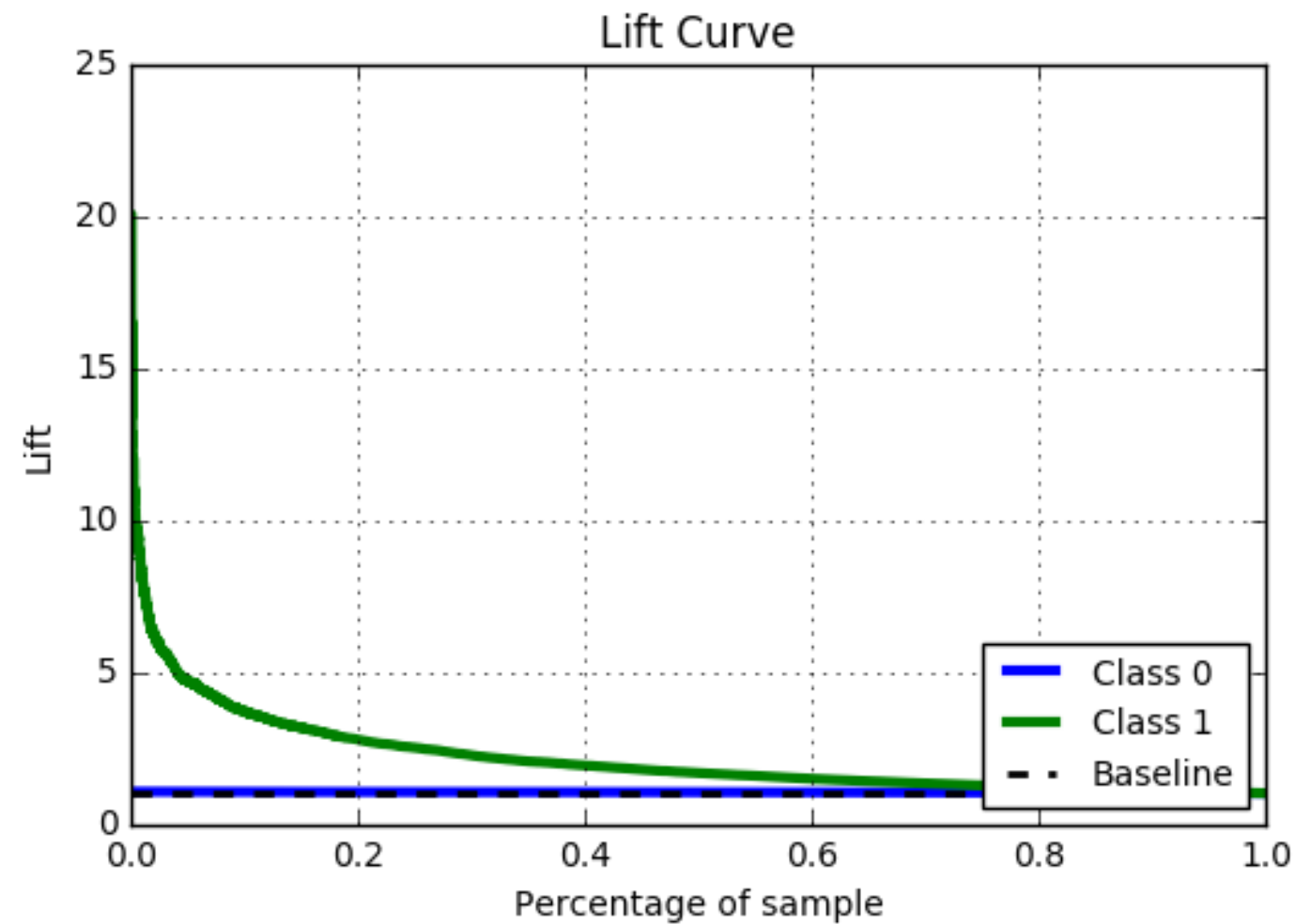
# The lift curve in Python

```
import scikitplot as skplt
import matplotlib.pyplot as plt

skplt.metrics.plot_lift_curve(true_values, predictions)
plt.show()
```



# The lift curve in Python





## FOUNDATIONS OF PREDICTIVE ANALYTICS IN PYTHON (PART 1)

**Let's practice!**



## FOUNDATIONS OF PREDICTIVE ANALYTICS IN PYTHON (PART 1)

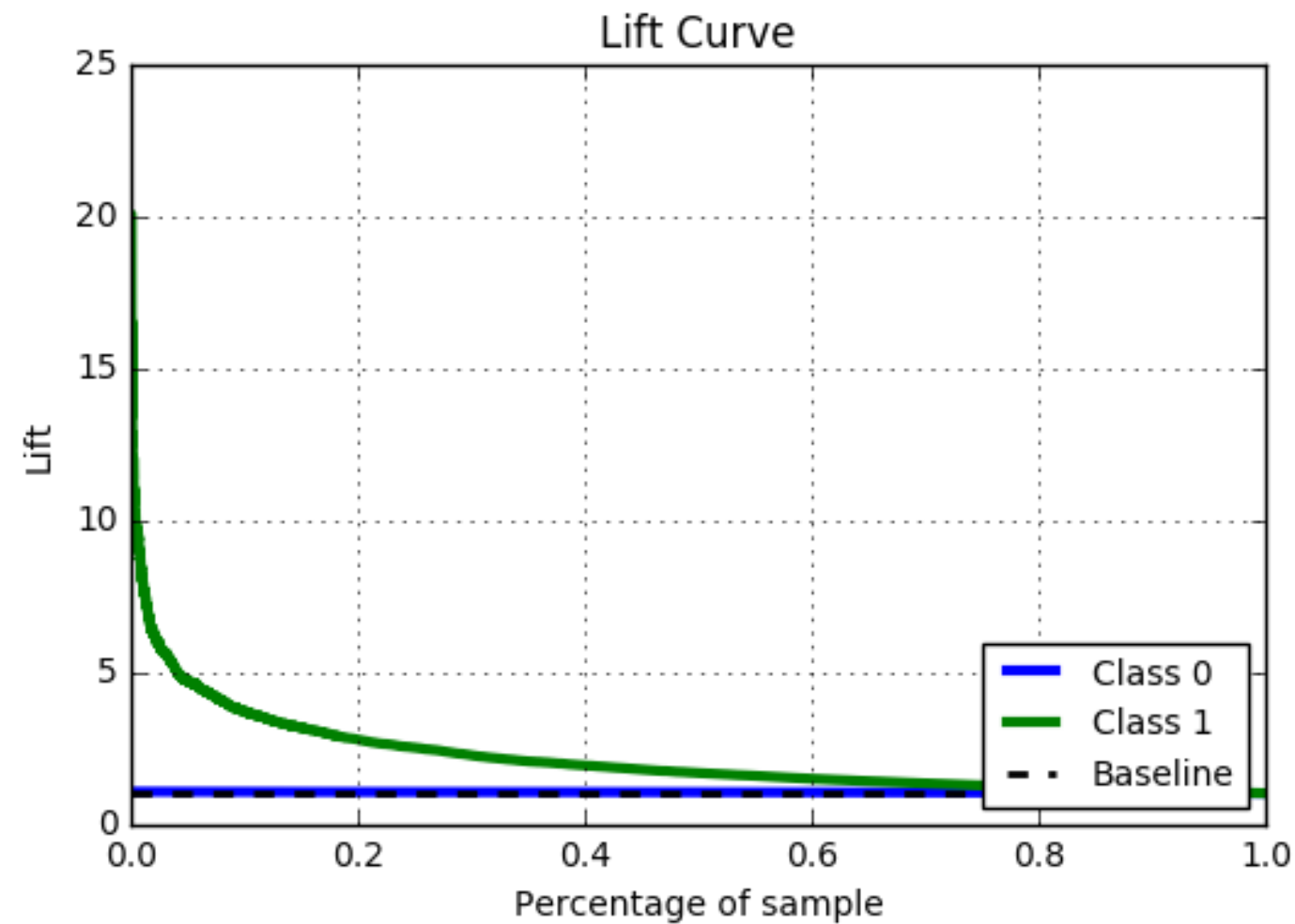
# Guiding business to better decisions

Nele Verbiest, Ph.D

Data Scientist  
Python Predictions



# Estimating profit





# Estimating profit

```
population_size = 100_000
target_incidence = 0.05
reward_target = 50
cost_campaign = 2

def profit(perc_targets, perc_selected, population_size, reward_target,
           cost_campaign)
    cost = cost_campaign * perc_selected *
           population_size
    reward = reward_target * perc_targets * perc_selected *
            population_size
    return(reward - cost)

perc_selected = 0.20
lift = 2.5
perc_targets = lift * target_incidence
print(profit(perc_targets, perc_selected, population_size,
             reward_target, cost_campaign))

60000

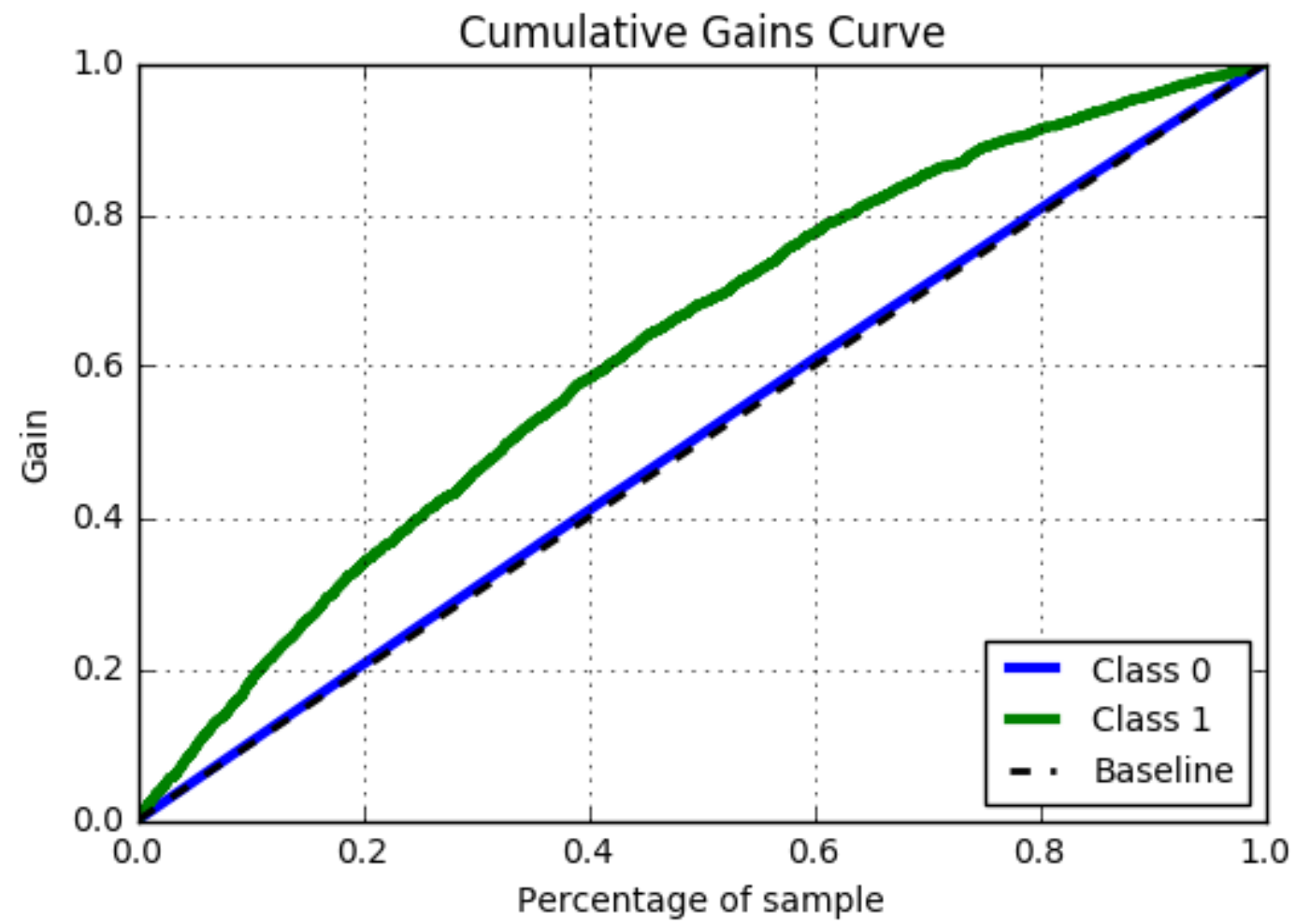
print(profit(target_incidence, 1, population_size, reward_target, cost_campaign))

-50000
```





# Campaign selection





# Campaign Selection

```
# Information about the campaign
population_size = 1000000
target_incidence = 0.02

# Number of targets you want to reach
number_targets_toreach = 16000
perc_targets = number_targets_toreach / (target_incidence * population_size)
print(perc_targets_toreach)

0.8

cumulative_gains = 0.60
# Number of donors to reach
number_donors_toreach = cumulative_gains * population_size
print(number_donors_toreach)

600 000
```



## FOUNDATIONS OF PREDICTIVE ANALYTICS IN PYTHON (PART 1)

**Let's practice!**