HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Introduction to HR analytics

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

What is HR analytics?

- Also known as People analytics
- Is a data-driven approach to managing people at work.

Problems addressed by HR analytics

- Hiring/Assessment
- Retention
- Performance evaluation
- Learning and Development
- Collaboration/team composition
- Other (e.g. absenteeism)

Employee turnover

- Employee turnover is the process of employees leaving the company
- Also known as employee attrition or employee churn
- May result in high costs for the company
- May affect company's hiring or retention decisions

Course structure

1. Describing and manipulating the dataset
2. Predicting employee turnover
3. Evaluating and tuning prediction
4. Selection final model

The Dataset

```
In  [1]: import pandas as pd
         data = pd.read_csv("turnover.csv")

In  [2]: data.info()

Out [2]:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14999 entries, 0 to 14998
Data columns (total 10 columns):
satisfaction_level       14999 non-null float64
last_evaluation          14999 non-null float64
number_project           14999 non-null int64
average_montly_hours     14999 non-null int64
time_spend_company       14999 non-null int64
work_accident            14999 non-null int64
churn                    14999 non-null int64
promotion_last_5years    14999 non-null int64
department               14999 non-null object
salary                   14999 non-null object
dtypes: float64(2), int64(6), object(2)
memory usage: 1.1+ MB
```

The Dataset (cont'd)

```
In [1]: data.head()
```

	satisfaction_level	last_evaluation	number_project	average_montly_hours	time_spend_company	work_accident	churn	promotion_last_5years	department	salary
0	0.38	0.53	2	157	3	0	1	0	sales	low
1	0.8	0.86	5	262	6	0	1	0	sales	medium
2	0.11	0.88	7	272	4	0	1	0	sales	medium
3	0.72	0.87	5	223	5	0	1	0	sales	low
4	0.37	0.52	2	159	3	0	1	0	sales	low

Unique values

```
In [1]: print(data.salary.unique())

array(['low', 'medium', 'high'], dtype=object)
```

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Transforming categorical variables

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

Types of categorical variables

- Ordinal - variables with two or more categories that can be ranked or ordered
  - Our example: **salary**
  - Values: low, medium, high
- Nominal - variables with two or more categories with **do not** have an instrinsic order
  - Our example: **department**
  - Values: sales, accounting, hr, technical, support, management, IT, product_mng, marketing, RandD

Encoding categories (salary)

```
In [1]: # Change the type of the "salary" column to categorical
        data.salary = data.salary.astype('category')

In [2]: # Provide the correct order of categories
        data.salary = data.salary.cat.reorder_categories(['low',
                                                          'medium',
                                                          'high'])

In [3]: # Encode categories with integer values
        data.salary = data.salary.cat.codes
```

Old values	New values
low	0
medium	1
high	2

Getting dummies

```
In [1]: # Get dummies and save them inside a new DataFrame
        departments = pd.get_dummies(data.department)
```

Example output

IT	RandD	accounding	hr	management	marketing	product_mng	sales	support	technical
0	0	0	0	0	0	0	0	0	1

Dummy trap

```
In [1]: departments.head()
```

IT	RandD	accounding	hr	management	marketing	product_mng	sales	support	technical
0	0	0	0	0	0	0	0	0	1

```
In [1]: departments = departments.drop("technical", axis = 1)
In [2]: departments.head()
```

IT	RandD	accounding	hr	management	marketing	product_mng	sales	support
0	0	0	0	0	0	0	0	0

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Descriptive Statistics

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

Turnover rate

```
In [1]: # Get the total number of observations and save it
        n_employees = len(data)

In [2]: # Print the number of employees who left/stayed
        print(data.churn.value_counts())

In [3]: # Print the percentage of employees who left/stayed
        print(data.churn.value_counts()/n_employees*100)

Out [3]:

0    76.191746
1    23.808254
Name: churn, dtype: float64
```

Summary

Stayed	Left
76.19%	23.81%

Correlations

```
In [1]: import matplotlib.pyplot as plt
In [2]: import seaborn as sns
In [3]: corr_matrix = data.corr()
In [4]: sns.heatmap(corr_matrix)
In [5]: plt.show()
```

	satisfaction_level	last_evaluation	number_project	average_montly_hours	time_spend_company	work_accident	churn	promotion_last_5years	salary
satisfaction_level	1	0.11	-0.14	-0.02	-0.10	0.06	-0.39	0.03	0.05
last_evaluation	0.11	1	0.35	0.34	0.13	-0.01	0.01	-0.01	-0.01
number_project	-0.14	0.35	1	0.42	0.20	0.00	0.02	-0.01	0.00
average_montly_hours	-0.02	0.34	0.42	1	0.13	-0.01	0.07	0.00	0.00
time_spend_company	-0.10	0.13	0.20	0.13	1	0.00	0.14	0.07	0.05
work_accident	0.06	-0.01	0.00	-0.01	0.00	1	-0.15	0.04	0.01
churn	-0.39	0.01	0.02	0.07	0.14	-0.15	1	-0.06	-0.16
promotion_last_5years	0.03	-0.01	-0.01	0.00	0.07	0.04	-0.06	1	0.10
salary	0.05	-0.01	0.00	0.00	0.05	0.01	-0.16	0.10	1

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Splitting the data

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

Target and features

- target = churn
- features = everything else

Train/test split

- train - the component used to develop the model
- test - the component used to validate the model

```
from sklearn.model_selection import train_test_split

target_train, target_test, features_train, features_test =
                              train_test_split(target,features,test_size=0.25)
```

Overfitting

an error that occurs when model works well enough for the dataset it was developed on (train) but is not useful outside of it (test)

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Introduction to Decision Tree classification

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

Classification in Python

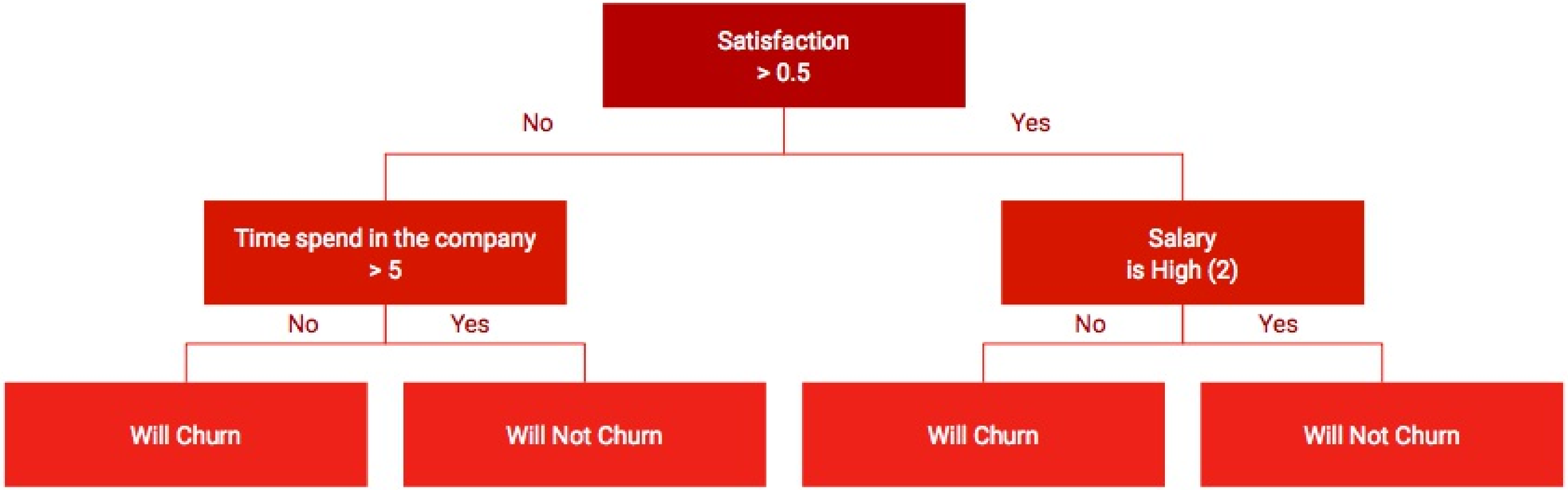
Classification algorithms

- Logistic regression
- Support Vector Machines
- Neural Networks
- Other algorithms

Algorithm we will use

- Decision Tree

Decision Tree Classification



Splitting rule

Splitting rules:

- Gini: 2*p*(1-p)
- Entropy: -p*log(p) - (1-p)*log(1-p)

Decision Tree splitting: hypothetical example

Total set: 100 observations, 40 left, 60 stayed

- Gini: 2*0.4*0.6 = 0.48

Splitting rule: satisfaction > 0.8

- Left branch (YES) - 50 people: all stayed
- Gini: 2*1*0 = 0
- Right branch (NO) - 50 people: 40 left, 10 stayed
- Gini: 2*0.4*0.1 = 0.08

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Predicting employee churn using decision trees

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

Decision Tree in Python

```
from sklearn.tree import DecisionTreeClassifier

model = DecisionTreeClassifier(random_state=42)

model.fit(features_train,target_train)

model.score(features_test,target_test)*100
```

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

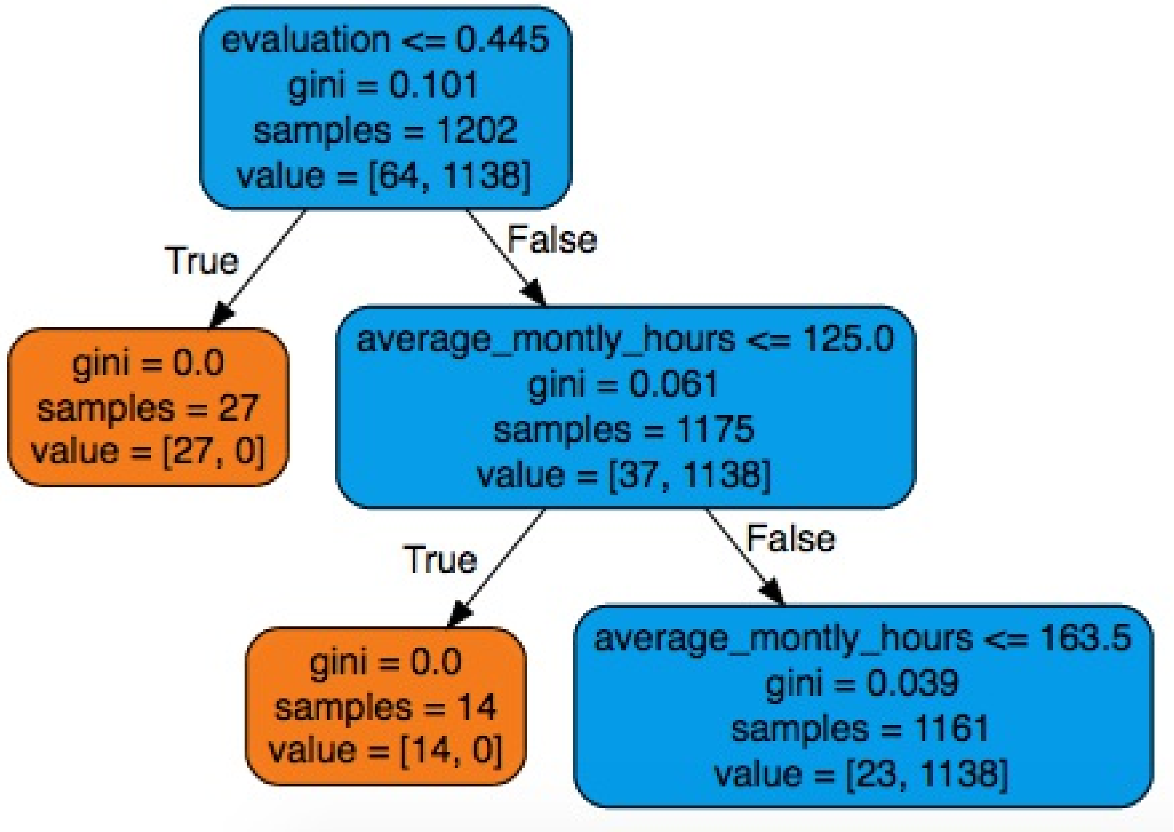
Interpretation of the decision tree

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

Visualization

1. Export
2. Copy content
3. Paste it in www.webgraphviz.com

Interpretation



HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Tuning employee turnover classifier

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

Overfitting

Existance of overfitting:

- Training accuracy: 100%
- Testing accuracy: 97.23%

Methods to fight it:

- Limiting tree maximum depth
- Limiting minimum saple size in leafs

Pruning the tree

Limiting Depth

```
model_depth_5 = DecisionTreeClassifier(
                max_depth=5, random_state=42)

# Train set Accuracy: 97.71%
# Test  set Accuracy: 97.06%
```

Limiting Samples

```
model_sample_100 = DecisionTreeClassifier(
                   min_samples_leaf=100, random_state=42)

# Train set Accuracy: 96.58%
# Test set Accuracy: 96.13%
```

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Evaluating the model

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

Prediction errors

Confusion Matrix		Reality	
		0	1
Predicted	0	TN	FN
	1	FP	TP

Evaluation metrics 1

- If target is leavers, focus on FN
  - Recall score = TP/(TP+FN)
  - Lower FN, higher Recall score
  - Recall score - % of correct predictions among 1s (leavers)
- If target is stayers, focus on FP
  - Specificity = TN/(TN+FP)
  - Lower FP, higher Specificity,
  - Specificity - % of correct predictions among 0s (stayers)

Evaluation metrics 2

- Even if target is leavers, you may still focus on FP:
  - Precision score = TP/(TP+FP)
  - Lower FP, higher Recall score
  - Precision score - % of leavers in reality, among those predicted to leave

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

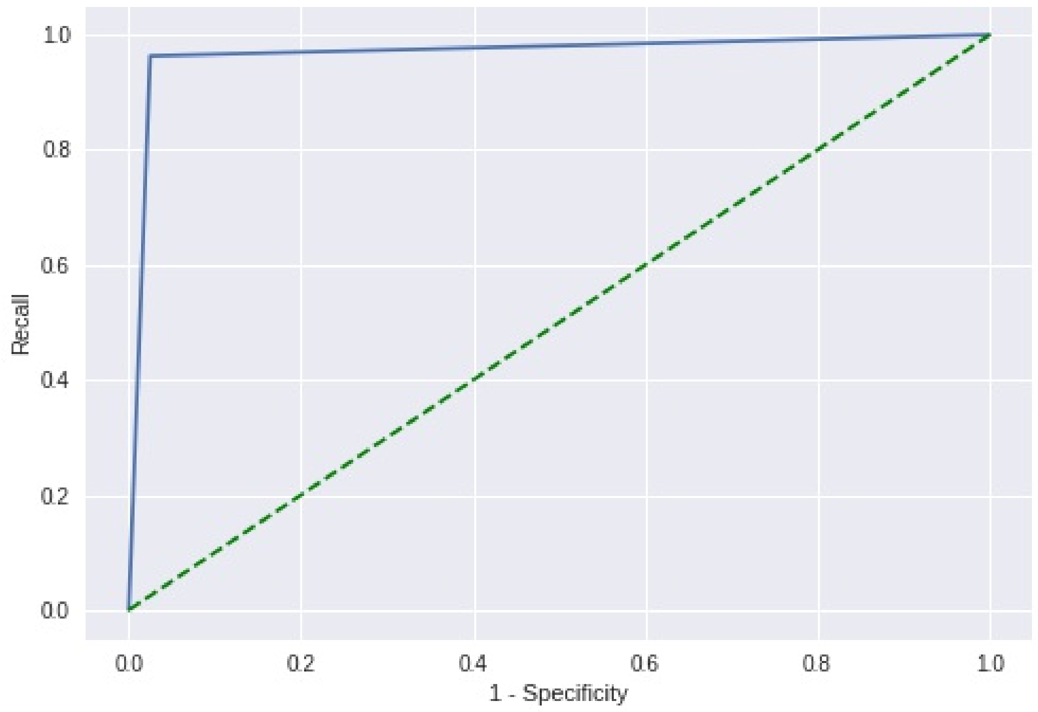
Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Targeting both leavers and stayers

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

AUC score



- Vertical axis: Recall
- Horizontal axis: 1 - Specificity
- Blue line: ROC
- Green line: baseline
- Area between blue and green: AUC

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Class Imbalance

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

Prior probabilities

Without balance

- $P_0 = 0.76$
- $P_1 = 0.24$
- $Gini = 0.36$

With balance

- $P_0 = 0.5$
- $P_1 = 0.5$
- $Gini = 0.5$

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Hyperparameter tuning

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

GridSearch

Values for minimum samples in the leaf

Values for maximum depth

	50	100	150	200	250	300	350	400	450
5	5, 50	5, 100	5, 150	5, 200	5, 250	5, 300	5, 350	5, 400	5, 450
6	6, 50	6, 100	6, 150	6, 200	6, 250	6, 300	6, 350	6, 400	6, 450
7	7, 50	7, 100	7, 150	7, 200	7, 250	7, 300	7, 350	7, 400	7, 450
8	8, 50	8, 100	8, 150	8, 200	8, 250	8, 300	8, 350	8, 400	8, 450
9	9, 50	9, 100	9, 150	9, 200	9, 250	9, 300	9, 350	9, 400	9, 450
10	10, 50	10, 100	10, 150	10, 200	10, 250	10, 300	10, 350	10, 400	10, 450
11	11, 50	11, 100	11, 150	11, 200	11, 250	11, 300	11, 350	11, 400	11, 450
12	12, 50	12, 100	12, 150	12, 200	12, 250	12, 300	12, 350	12, 400	12, 450
13	13, 50	13, 100	13, 150	13, 200	13, 250	13, 300	13, 350	13, 400	13, 450
14	14, 50	14, 100	14, 150	14, 200	14, 250	14, 300	14, 350	14, 400	14, 450
15	15, 50	15, 100	15, 150	15, 200	15, 250	15, 300	15, 350	15, 400	15, 450
16	16, 50	16, 100	16, 150	16, 200	16, 250	16, 300	16, 350	16, 400	16, 450
17	17, 50	17, 100	17, 150	17, 200	17, 250	17, 300	17, 350	17, 400	17, 450
18	18, 50	18, 100	18, 150	18, 200	18, 250	18, 300	18, 350	18, 400	18, 450
19	19, 50	19, 100	19, 150	19, 200	19, 250	19, 300	19, 350	19, 400	19, 450
20	20, 50	20, 100	20, 150	20, 200	20, 250	20, 300	20, 350	20, 400	20, 450

Cross-Validation

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Test	Train	Train	Train	Train
Train	**Test**	Train	Train	Train
Train	Train	**Test**	Train	Train
Train	Train	Train	**Test**	Train
Train	Train	Train	Train	**Test**

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Important features for predicting attrition

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

Feature Importances

- Importance is calculated as relative decrease in Gini due to the selected feature.
- Importances are scaled to sum up to 100%.
- Higher percentage, higher importance.

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Let's practice!

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

Final thoughts

Hrant Davtyan
Assistant Professor of Data Science
American University of Armenia

Alternative methods

- Logistic Regression
- Tree based
  - Random Forest
  - Gradient Boosting
- Neural Networks

HR ANALYTICS IN PYTHON: PREDICTING EMPLOYEE CHURN

The End