Introduction to Data Science

Case Description

Prof. Dr. Hendrik Meth
HdM, Stuttgart

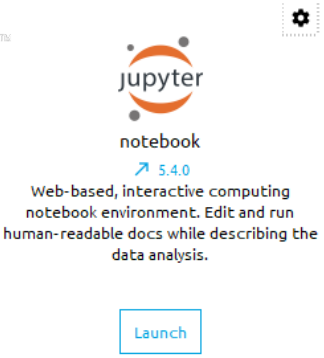
Course Grading

Element	Description	Exam / Due Date

Project (3 teams of 4)	• Analysis case study to be solved with Python in teams
	• Teams can be chosen by yourself
		Due Dates:
		• WP1 Data Exploration
		  • Moodle upload / GitHub Freeze: 16.11.2023
		  • Presentation: 17.11.2023
		• WP2 Data Preparation
		  • Moodle upload / GitHub Freeze: 21.12.2023
		  • Presentation: 22.12.2023
		• WP3 Modeling & Validation
		  • Moodle upload / GitHub Freeze: 25.1.2024
		  • Presentation: 26.1.2024

Python Labs - Infrastructure

Student notebooks



Local files

HOCHSCHULE DER MEDIEN

Picture Source: Pixabay

Dataset

- Structured dataset describing bike sharing data
- Method to predict label: Regression
- Format:
  - 732 rows × 18 columns
  - 15 features
  - 3 labels: Main label to be used is “cnt” (number of bike rentals per day)

Picture: pixabay

Dataset

#	Attribute	Description
1	instant	record index
2	dteday	date
3	season	season (1:winter, 2:spring, 3:summer, 4:fall)
4	yr	year
5	mnth	month (1 to 12)
6	holiday	whether day is holiday or not
7	weekday	day of the week
8	workingday	if day is neither weekend nor holiday is 1, otherwise is 0.
9	weathersit	weather situation: 1: Clear, Few clouds, Partly cloudy, Partly cloudy 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
10	temp	Temperature
11	atemp	Feeling temperature
12	hum	Humidity
13	windspeed	Wind speed
14	leaflets	Number of distributed leaflets (for marketing purposes)
15	price reduction	Day with price reduction (yes = 1, no =0)
16	casual	count of casual users
17	registered	count of registered users
18	cnt	count of total rental bikes including both casual and registered

Task 1: Data Exploration

- Explore the provided data sets using descriptive statistics (e.g. mean values, standard deviations, min/max values, missing values) and visualizations (e.g. histograms, boxplots)
- Point out which data quality issues you identified in terms of
  - Missing values
  - Outliers
  - Features to be transformed (e.g. normalization) transformation
  - Features to be removed (feature selection)
  - Other insights which require attention in the following phases

Task 2: Model Baseline and Data Preparation

- Create a baseline linear regression model using the originally provided training dataset with minimal preprocessing and evaluate it with your test dataset based on accuracies (MAE) and *coefficient of determination* (R^2)
- Preprocess the original datasets to address the identified data quality issues
  - Missing values
  - Outliers
  - Features to be transformed (e.g. normalization)
  - …
- Within the **test** dataset, you are not allowed to:
  - Remove examples / rows
  - Change label values
- Create a new linear regression model using the preprocessed training dataset and evaluate it with your preprocessed test dataset based on *accuracies* (MAE) and *coefficient of determination* (R^2)
- Export the pre-processed training and test dataset to a CSV

Task 3: Modeling Optimization

- Create a new baseline linear regression model using the preprocessed training dataset and evaluate it with your preprocessed test dataset based on *accuracies* (MAE) and *coefficient of determination* (R^2)
- Optimize your model / create further model versions
  - Algorithm Selection: Experiment with different regression algorithms, e.g. linear regression, polynomial regression, regression trees etc.
  - Hyper-parameter Tuning: Change the hyper-parameters of your algorithms (e.g. „degree“ in case of polynomial regression)
  - Feature Selection: Remove features, which are not helpful for your model
- Evaluate each model version based on accuracies (MAE) and *coefficient of determination* (R^2) using the test data and store evaluation results in a data frame
- Create an overview of your evaluation data frame…
  - By generating an overview table
  - By generating an appropriate visualization

Task Presentations

For each task:

- 10-15 mins presentation
- Structure your presentation based on the bullet points of the task description
- Upload one zip in Moodle containing:
  - One Jupyter Notebook per presentation
  - The data files used in your Jupyter notebook
  - One Powerpoint file per presentation
- Every team member presents his/her part

Task Presentations

- Jupyter Notebook
  - Should contain extensive comments and markup to structure and explain the coding
  - Should be uploaded in executed state (all result cells visible)
  - Should be uploaded together with the data files you used
- Powerpoint presentation
  - Should not contain coding
  - Should focus on result cells of your Jupyter notebook (tables, diagrams etc.)
  - Focus on interpretation and analysis of results

Questions

?

Project How To

Task 1: Data Exploration

- Explore the provided data sets using descriptive statistics (e.g. mean values, standard deviations, min/max values, missing values) and visualizations (e.g. histograms, boxplots)
- Point out which data quality issues you identified in terms of
  - Missing values
  - Outliers
  - Features to be transformed (e.g. normalization) transformation
  - Features to be removed (feature selection)

How To - Task 1

- Explore the provided data sets using descriptive statistics (e.g.
-

Wichtig:

- Die Exploration wird für Trainings- und Testdaten gemeinsam durchgeführt
- Die Programmierung dient hier nur als Mittel zum Zweck. Jupyter Notebooks werden als separate Abgabe, zusätzlich zur Präsentation abgegeben
- Jupyter Notebooks immer im ausgeführten Zustand (mit Result Cells) abgeben!
- Kein Programm-Code in der Präsentation
- In der Präsentation sind die erzeugten Diagramme und Statistiken sowie insbesondere deren Analyse wichtig
- Erkenntnisse aus der Analyse sollten daher in Bullet Points oder Fließtext auf den Folien mit den Diagrammen formuliert werden
- Es sollte am Ende von Task 1 eine Zusammenfassung erstellt werden, aus der ersichtlich wird, welche Data Preparation-Schritte in Task 2 angegangen werden und warum (Begründung durch Erkenntnisse aus Task 1)

Example - Task 1

Data Tab

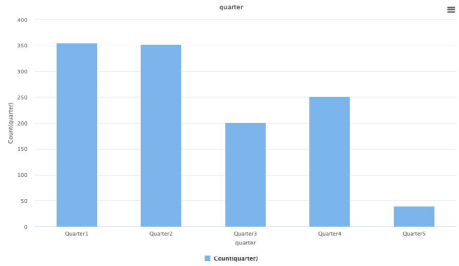
Bei *no_of_workers und smv* stellt man fest, dass sich die Werte kaum bis gar nicht überschneiden (mit Ausnahme der Outlier bei *no_of_workers*).

In Hinblick auf die Missing Values bei *wip* und den Nullwerten der vorherigen Folie lädt diese Auffälligkeit umso mehr dazu ein, die Daten anhand von *department* aufzuteilen.

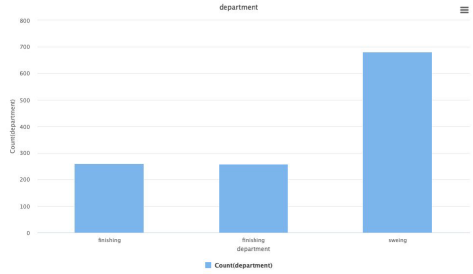
department	no_of_w...
sweing	30.500
sweing	30.500
sweing	30.500
sweing	30.500
sweing	30.500
sweing	29.500
finishing	28
sweing	27
sweing	27
sweing	26
finishing	25
finishing	25
finishing	25
finishing	25
finishing	25
finishing	25
finishing	25
finishing	25

department	smv
sweing	11.410
sweing	11.410
sweing	11.410
sweing	10.050
sweing	10.050
sweing	10.050
sweing	10.050
sweing	10.050
sweing	10.050
sweing	10.050
finishing	5.130
finishing	5.130
finishing	5.130
finishing	5.130
finishing	5.130
finishing	5.130
finishing	5.130
finishing	5.130

Data Exploration: Quarter & Department



Laut der Dataset Beschreibung beschreibt das Attribut „quarter" die Unterteilung eines Monatszeitraums in vier gleiche Quartale. Ein Quartal beschreibt hier einen Zeitraum von 7 bis 8 Tagen. Bei der Data Exploration ist aufgefallen, dass es Werte zu einem „5. Quartal" gibt. Quartale bestehen jedoch nur aus vier Abschnitten. Da die Werte des „5. Quartals" im Zeitraum vom 29. bis zum 31. eines Monats liegen werden wir für die weitere Analyse die Werte des 5. Quartals mit den Werten des 4. Quartals zusammenführen.



Das Attribut „department" beschreibt laut Dataset Beschreibung die mit der Instanz assoziierte Abteilung. Hier fällt auf, dass es zwei gleichnamige Feature-Ausprägungen von „finishing" gibt. Es hat sich herausgestellt, dass eines der „finishing" Datensätze durch ein extra Leerzeichen vom Rapidminer Programm als ein anderes Feature erkannt wurde. Dies haben wir für die weitere Analyse korrigiert, sodass es für das Attribut „department" nun zwei Ausprägungen „sweing" und „finishing" gibt.

HOCHSCHULE DER MEDIEN

Task 2: Model Baseline and Data Preparation

- Create a baseline linear regression model using the originally provided training dataset with minimal preprocessing and evaluate it with your test dataset based on accuracies (MAE) and *coefficient of determination* (R^2)
- Preprocess the original datasets to address the identified data quality issues
  - Missing values
  - Outliers
  - Features to be transformed (e.g. normalization) transformation
- Within the **test** dataset, you are not allowed to:
  - Remove examples / rows
  - Change label values
- Create a new linear regression model using the preprocessed training dataset and evaluate it with your preprocessed test dataset based on *accuracies* (MAE) and *coefficient of determination* (R^2)
- Export the pre-processed training and test dataset to a CSV

How To - Task 2

- Crea
  d
-
-
-
-
-

Wichtig:

- Task2 sollte möglichst gut an Task1 anknüpfen, d.h. unmittelbar die Data Preparation-Schritte umsetzen, die in der Zusammenfassung von Task 1 empfohlen wurden
- Die Data Preparation wird für Trainings- und Testdaten getrennt durchgeführt
- Weder in Trainings- noch in Testdaten darf das Label geändert werden
- In den Testdaten dürfen keine Zeilen entfernt werden
- Die Effekte der einzelnen Data Preparation-Schritte sollten jeweils auch einzeln untersucht werden. Wichtig: das muss dokumentiert werden, d.h. die jeweiligen Modell-Genauigkeiten (MAE-Werte, R^2-Werte) sollten in ein Log-File weggeschrieben werden, incl. einer Beschreibung des Datensatzes (WAS wurde angepasst)
- Wichtig: Task2 und Task3 sind nicht reines „Trial and Error“, es sollte jeweils auch begründet werden, warum ein Prepping-Schritt durchgeführt wird (entweder basierend auf Task 1 oder neuer Erkenntnisse aus Task 2)

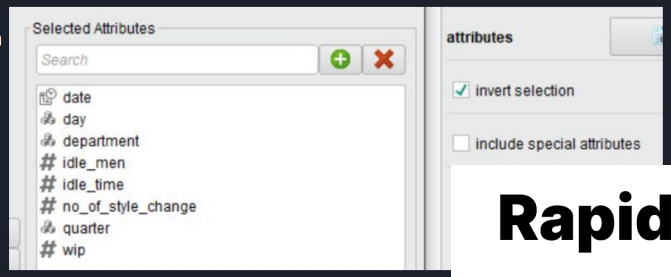
Example - Task 2

Data Preparation

Da es bei *finishing* für die Features *idle_men*, *idle_time*, *no_of_style_change* und *wip* keine oder nur Nullwerte gibt, haben wir diese im finishing Prozess entfernt.

Das Aufteilen der Departments hat natürlich auch zu Änderungen im MAE geführt.

Bei *sweing* ist dieser nun 0,129 und bei *finishing* 0,221.



Rapidminer Baseline

Nach dem erneuten Prozessdurchlauf entstanden die Modelle, welche in der Tabelle zu sehen sind. Vor allem die Modelle A und B sind gute Ansätze für die Modellweiterentwicklung in Python.

	Features	RMSE	R^2
Modell A	-Targeted-Productivity -SMV -Incentive -Idle Men	0.089	0.663
Modell B	-Targeted-Productivity -SMV -Over-Time -Incentive -Working-Condition F3	0.018	0.857
Modell C	-Targeted-Productivity -SMV -No. Of Workers	0.192	0.139
Modell D	-Targeted-Productivity -SMV -Overtime -No. Of Workers	0.045	0.123

Task 3: Modeling Optimization

- Create a new baseline linear regression model using the preprocessed training dataset and evaluate it with your preprocessed test dataset based on *accuracies* (MAE) and *coefficient of determination* (R^2)
- Optimize your model / create further model versions
  - Algorithm Selection: Experiment with different regression algorithms, e.g. linear regression, polynomial regression, regression trees etc.
  - Hyper-parameter Tuning: Change the hyper-parameters of your algorithms (e.g. „degree“ in case of polynomial regression)
  - Feature Selection: Remove features, which are not helpful for your model
- Evaluate each model version based on accuracies (MAE) using the test data and store evaluation results in a data frame
- Create an overview of your evaluation data frame…
  - By generating an overview table
  - By generating an appropriate visualization

How To: Task 3

- C
- By generating an appropriate visualization

Wichtig:

- Task3 sollte möglichst gut an Task2 anknüpfen, d.h. es werden nicht mehr die Original-Trainings- und Testdaten verwendet, sondern bereits die vorbereiteten Daten
- Weder in Trainings- noch in Testdaten darf das Label geändert werden
- In den Testdaten dürfen keine Zeilen entfernt werden
- Die Effekte der einzelnen Optimization-Schritte sollten jeweils auch einzeln untersucht werden. Wichtig: das muss dokumentiert werden, d.h. die jeweiligen Modell-Genauigkeiten (MAE-Werte) sollten in ein Log-File weggeschrieben werden, incl. einer Beschreibung der Optimization / des Modells
- Hier sollte auch mit Diagrammen gearbeitet werden um das Ergebnis zu verdeutlichen
- Wichtig: Die Ergebnisse sollten auch analysiert und diskutiert werden, d.h. in der Präsentation nicht nur Modell-Varianten und –Genauigkeiten darstellen, sondern auch interpretieren und erklären

Example - Task 3

TASK 3: Modelling optimization

- Wir haben die csv-Dateien von Rapidminer in den Code in Jupyter Notebook eingefügt, um so den niedrigsten MAE Wert herauszufinden. Hierfür haben wir die Regression Tree, lineare-, polynomiale- und KNN-Regression angewendet.
- Mit dem Nutzen und durchtesten der Daten der folgenden Features *team, targeted_productivity, smv, idle_men, no_of_style_change* haben wir die besten Ergebnisse für „MAE finishing & sweing" erzielt.
- Mit dem Nutzen und durchtesten der Daten der folgenden Features *team, targeted_productivity, smv, overtime, idle_time, no_of_style_change* haben wir die besten Ergebnisse für „MAE sweing, best correalation" erzielt.
- Mit dem Nutzen und durchtesten der Daten der folgenden Features *team, targeted_productivity, smv, working_cond_f1* haben wir die besten Ergebnisse für „MAE finishing, best correalation" erzielt.
- Die Ergebnisse sind auf der nächsten Folie zu sehen.

odeling Optimization

Die Modelle haben also nun folgende Ergebnisse:

	Algorithmus mit den besten Ergebnissen	MAE	RMSE	R^2
Modell A	Linear Regression	0.058	0.085	0.735
Modell B	Regression Decision Tree	0.005	0.008	0.968
Modell C	K-NN	0.130	0.169	0.268
Modell D	K-NN	0.025	0.033	0.293

HOCHSCHULE DER MEDIEN

Questions

?