数据科学家直通车 Phase 2 - Week 1 实战课

By: Vincent

BITTIGER

The Lifelong Learning Platform of Silicon Valley

Outline

BITTIGER

- Feature Engineering
  - Feature Engineering Methods
  - Data wrangling with Uber Churn Data (Pandas)
    - Exploratory data analysis (EDA)
    - Imputation
    - Categorical variable to
    - Label creation
    - Prepare training data
- Supervised Learning Methods Hands-on (sklearn)
  - Implement supervised machine learning methods with sklearn
  - Model selection and cross-validataion
  - Grid search for hyper-parameter tuning

1. Feature Engineering and Data Wrangling

BITTIGER

The Lifelong Learning Platform of Silicon Valley

Feature Engineering

BITTIGER

- Decompose categorical attributes, e.g., country
- Feature binarization, generate binary feature: is_US Feature
- Discretization, aggregate countries by continent
- Keep countries of large bid volume, aggregate the rest
- Decompose date-time attributes
- Extract month/day/hour component
- Break continuous time into time periods: morning, afternoon, evening
- Extract pattern/seasonality as features
- Reframe numerical quantities
- Value transformation, e.g., standardization, log()
- Generate rate type: amount per unit time
- Extract summary stats: min, max, mean, median, sd ...

Feature Engineering

BITTIGER

- Encoding Categorical Features
- Binning Numerical Features
- Imputation of Missing Values
- Handling Outliers
- Value Rescaling
- Min-Max Scaling
- Standardization
- Normalization

Encoding Categorical Features

BITTIGER

- Often features are not given as continuous/quantitative values but categorical/qualitative
- For example, a person could have features
  - Gender: ["male", "female"]
  - Country: ["from Europe", "from US", "from Asia"]
  - Browser: ["uses Firefox", "uses Chrome", "uses Safari", "uses Internet Explorer"]
  - For a person instance
  - ["male", "from US", "uses Internet Explorer"] could be expressed as [0, 1, 3] (Ok for tree methods, but not for logistic regression)
- One hot encoding:
  - Gender=[1,0]; Region=[0,1,0]; Browser=[0,0,0,1]
  - N-1 dimension is enough, so: gender=[1]; Region=[0,1]; Browser=[0,0,0]

Binning Numerical Features

BITTIGER

- Sometimes we also want to convert continuous variable to a categorical variable
- Example :Test score: [25, 94, 57, 62, 70, 25, 94, 57, 62, 70, 62, 70]
  - Define bins as **0 to 25**, **25 to 50**, **50 to 75**, **75 to 100**
  - Create names for the four groups [Low, Okay, Good, Great]
  - Bin the test score as [Low, Great, Good, Good, Good, Low, Great, Good, Good, Good, Good, Good]

Imputation of Missing Values

BITTIGER

- Cause for missing value
- Data collection
- Missing at random or not
- Data extraction
- Deal with missing value
- Deletion (be careful)
- Mean/Mode/Median imputation
- Prediction model imputation
- KNN (nearest neighbor) imputation
- + missing indicator: if imputed, missing=indicator

Handling Outliers

BITTIGER

- Outlier/spike data point can heavily influence ML models
- Trimming, truncating (remove outlier record, be careful)
- Simple procedure to exclude outliers/spikes
- Winsorizing, clipping, clamping, capping
- Limiting extreme values in the statistical data to reduce the effect of possibly spurious outliers
- Typical strategy is to set all outliers to a specified percentile of the data

Min-Max Rescaling

BITTIGER

- Min-Max Rescaling is to scale features to lie between a given minimum and maximum value, often between zero and one.
- The common method is rescaling the range of features to scale the range in [0, 1] or [−1, 1].

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Normalization

BITTIGER

- Normalization is the process of scaling individual samples to have unit norm. This process can be useful if you plan to use a quadratic form such as the dot-product or any other kernel to quantify the similarity of any pair of samples.

$$x' = \frac{x}{||x||}$$

More Preprocessing in Scikit-learn

BITTIGER

4.3. Preprocessing data

- 4.3.1. Standardization, or mean removal and variance scaling
  - 4.3.1.1. Scaling features to a range
  - 4.3.1.2. Scaling sparse data
  - 4.3.1.3. Scaling data with outliers
  - 4.3.1.4. Centering kernel matrices
- 4.3.2. Normalization
- 4.3.3. Binarization
  - 4.3.3.1. Feature binarization
- 4.3.4. Encoding categorical features
- 4.3.5. Imputation of missing values
- 4.3.6. Generating polynomial features
- 4.3.7. Custom transformers

Data wrangling with Uber Churn Data (Pandas)

BITTIGER

- Exploratory data analysis (EDA)
  - Counts, rates, groupby
  - Plot histograms (distributions)
  - With/without label
  - Matplotlib and data visualization
- Imputation
  - Drop
  - Fill with values: fixed, mean, median ...
- Categorical variable to numerical
  - One hot, get_dummies
- Time feature extraction: Hour of day, Day of week ...
- Label creation
- Prepare training data

Location: 1-Uber_Case_Study_ML_Demo/Uber_Rider_Case_Study_-_Data_Wrangling.ipynb

2. Supervised Learning Methods Hands-on

BITTIGER

The Lifelong Learning Platform of Silicon Valley

Supervised Learning Methods Hands-on (sklearn)

BITTIGER

- Implement supervised machine learning methods with sklearn
  - Decision Trees
  - KNN
  - Logistic Regression
  - Bagging
    - Bagged Trees
    - Bagged KNN
  - Random Forest
  - Gradient Boosting
  - SVM
  - Neural Network
- Model selection and cross-validataion
  - Evaluate overfitting with cross-validataion
  - Hand-on model tuning for each algorithm
  - Understand effect of major hyper-parameters on model performance
- Grid search for hyper-parameter tuning

Location: 1-Uber_Case_Study_ML_Demo/Uber_Rider_Churn_Supervised_Learning.ipynb

Summary

BITTIGER

- Feature Engineering
  - Feature Engineering Methods
  - Data wrangling with Uber Churn Data (Pandas)
    - Exploratory data analysis (EDA)
    - Imputation
    - Categorical variable to
    - Label creation
    - Prepare training data
- Supervised Learning Methods Hands-on (sklearn)
  - Implement supervised machine learning methods with sklearn
  - Model selection and cross-validataion
  - Grid search for hyper-parameter tuning