Feature Engineering

Papangkorn Inkeaw, Ph.D.

Explore Variables in Dataset

Lab 1

Inspect variables

Dataset: Titanic dataset (https://github.com/datasciencedojo/datasets/blob/master/titanic.csv)

- Import libraries

  import pandas as pd

  import numpy as np

  import matplotlib.pyplot as plt

- Read titanic dataset

  data = pd.read_csv('titanic.csv')

- Inspect the variable types

  data.dtypes

- Computer statistical descriptors

  data.describe()

Inspect variables

- Inspect the distinct values of a discrete variable

  data['Sex'].unique()

- Make a histogram for a numerical variable by dividing the variable value range into *m* intervals

  data['Age'].hist(bins=20)

Determine cardinality in categorical variables

- Count the number of unique categories in each variable

  data.nunique()

  data[['Sex', 'SibSp']].nunique()

- Plot the cardinality of each variable:

  data[['Sex', 'SibSp']].nunique().plot.bar(figsize=(12,6))

  plt.ylabel('Number of unique categories')

  plt.xlabel('Variables')

  plt.title('Cardinality')

Determine rare categories in categorical variables

- Display the unique categories of a variable

  data['Pclass'].unique()

- Calculate the number of sample per category and then divide them by the total number of sample in the dataset to obtain the percentage of sample per category.

  label_freq = data['Pclass'].value_counts() / len(data)

  print(label_freq)

- Make a bar plot showing the frequency of each category and highlight the 5% mark with a red line

  fig = label_freq.sort_values(ascending=False).plot.bar()

  fig.axhline(y=0.05, color='red') #make a red line of 5%

  fig.set_ylabel('percentage of sample within each category')

  fig.set_xlabel('Variable: Pclass')

  fig.set_title('Identifying Rare Categories')

  plt.show()

Identify a linear relationship between numerical variables

In this example, we use synthesized data for demonstration.

- Import libraries

  import pandas as pd

  import numpy as np

  import matplotlib.pyplot as plt

  import seaborn as sns

  from sklearn.linear_model import LinearRegression

  import scipy.stats as stats

- Create an x variable with 200 observations that are normally distributed

  np.random.seed(29)

  x = np.random.randn(200)

- Create a y variable that is linearly related to x with some added random noise

  *y = x * 10 + np.random.randn(200) * 2*

Identify a linear relationship between numerical variables

- Create a dataframe with the x and y variables

  data = pd.DataFrame([x, y]).T

  data.columns = ['x', 'y']

- Plot a scatter plot to visualize the linear relationship

  sns.lmplot(x="x", y="y", data=data, order=1)

  plt.ylabel("Target")

  plt.xlabel("Independent variable")

- Build a linear regression model between x and y

  linreg = LinearRegression()

  linreg.fit(data['x'].to_frame(), data['y'])

- Compute the coefficient of determination R^2

  score = linreg.score(data['x'].to_frame(), data['y'])

  print(score)

Identify a normal distribution

- Make a histogram and a density plot of the variable distribution

  sns.distplot(data['x'], bins=30)

- Create and display a Q-Q plot to assess a normal distribution

  stats.probplot(data['x'], dist="norm", plot=plt)

  plt.show()

Your work!

1. Download the Vehicle dataset from https://www.kaggle.com/datasets/nehalbirla/vehicle-dataset-from-cardekho
2. Explore the dataset using what you have learnt from this workshop
3. Export your python notebook (file format: .ipynb) and submit to assignment submission system (http://hw.cs.science.cmu.ac.th/)

Note:

- Put your name and student ID in the first cell using comment tag.
- Name your python notebook file with the pattern Lab_01_XXXXXXXXX.ipynb (XXXXXXXXX is your student ID)

References & Study Resources

- Soledad Galli. (2020). *Python Feature Engineering Cookbook*. Packt Publishing.
- https://github.com/datasciencedojo/datasets/blob/master/titanic.csv
- https://www.kaggle.com/datasets/nehalbirla/vehicle-dataset-from-cardekho