Module 2 – Part I

Setting up Deep Forecasting Environment (Python)

jupyter

TensorFlow

scikit learn

K Keras

ANACONDA

Visual Studio Code

Google colab

dmlc XGBoost

PROPHET

PYCARET

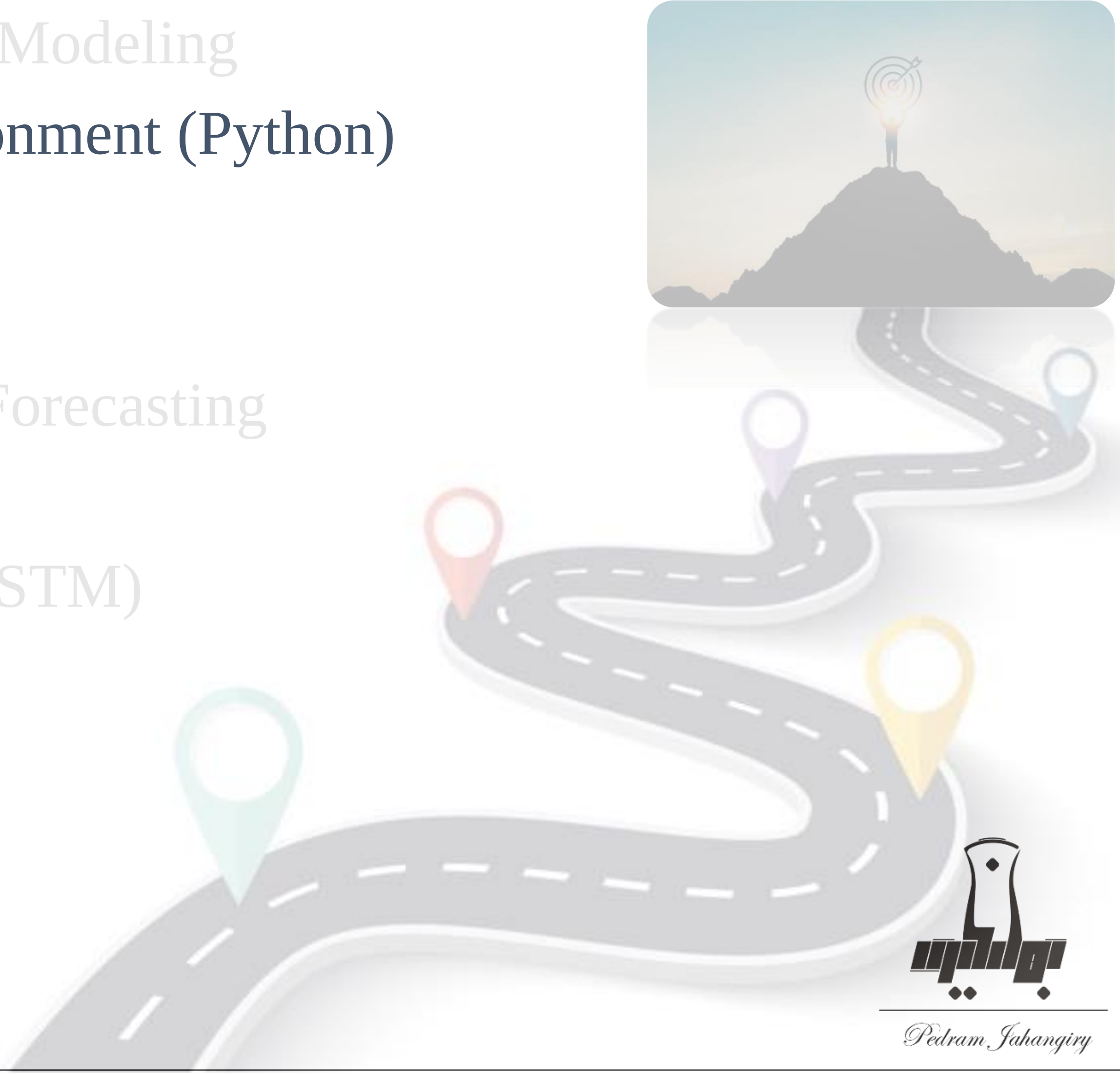
LightGBM

JON M. HUNTSMAN SCHOOL OF BUSINESS
UtahStateUniversity

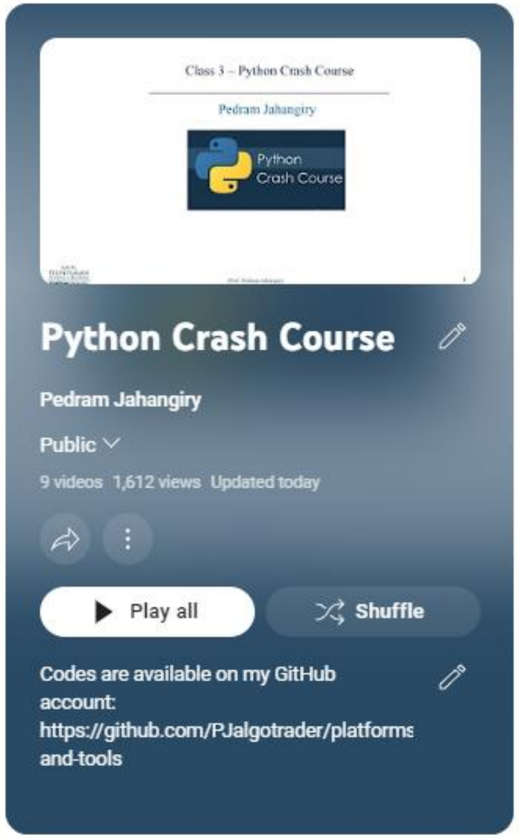
Pedram Jahangiry

Road map!

- Module 1- Demystifying Timeseries Data and Modeling
- Module 2- Setting up Deep Forecasting Environment (Python)
- Module 3- Exponential Smoothing
- Module 4- ARIMA models
- Module 5- Machine Learning for Time series Forecasting
- Module 6- Deep Neural Networks
- Module 7- Deep Sequence Modeling (RNN, LSTM)
- Module 8- Prophet and Neural Prophet



Course prerequisites



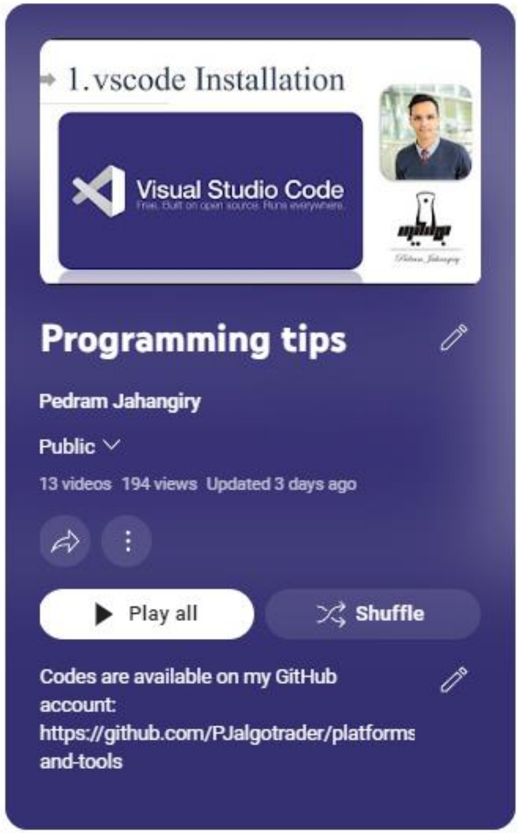
Python Crash Course

Pedram Jahangiry

Public

9 videos 1,612 views Updated today

Codes are available on my GitHub account: https://github.com/PJalgotrader/platforms-and-tools



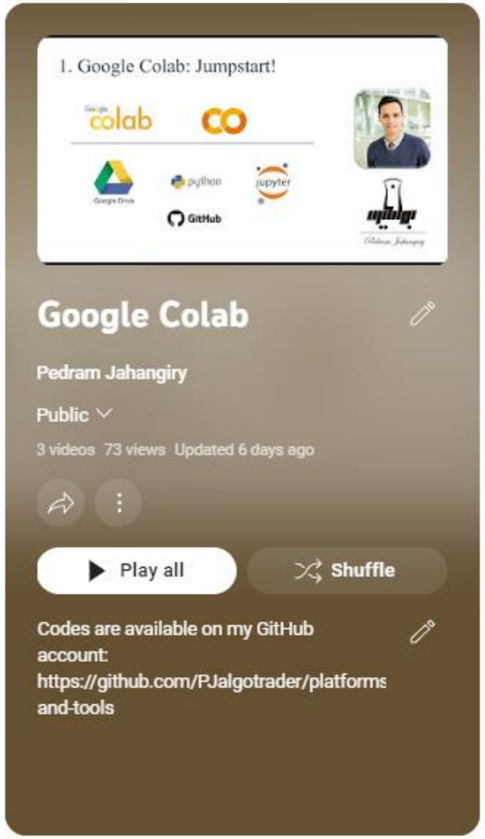
Programming tips

Pedram Jahangiry

Public

13 videos 194 views Updated 3 days ago

Codes are available on my GitHub account: https://github.com/PJalgotrader/platforms-and-tools



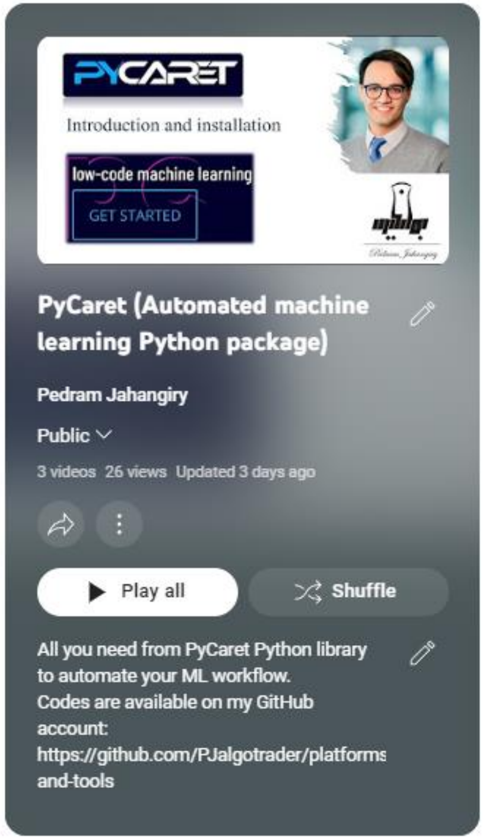
Google Colab

Pedram Jahangiry

Public

3 videos 73 views Updated 6 days ago

Codes are available on my GitHub account: https://github.com/PJalgotrader/platforms-and-tools



PyCaret (Automated machine learning Python package)

Pedram Jahangiry

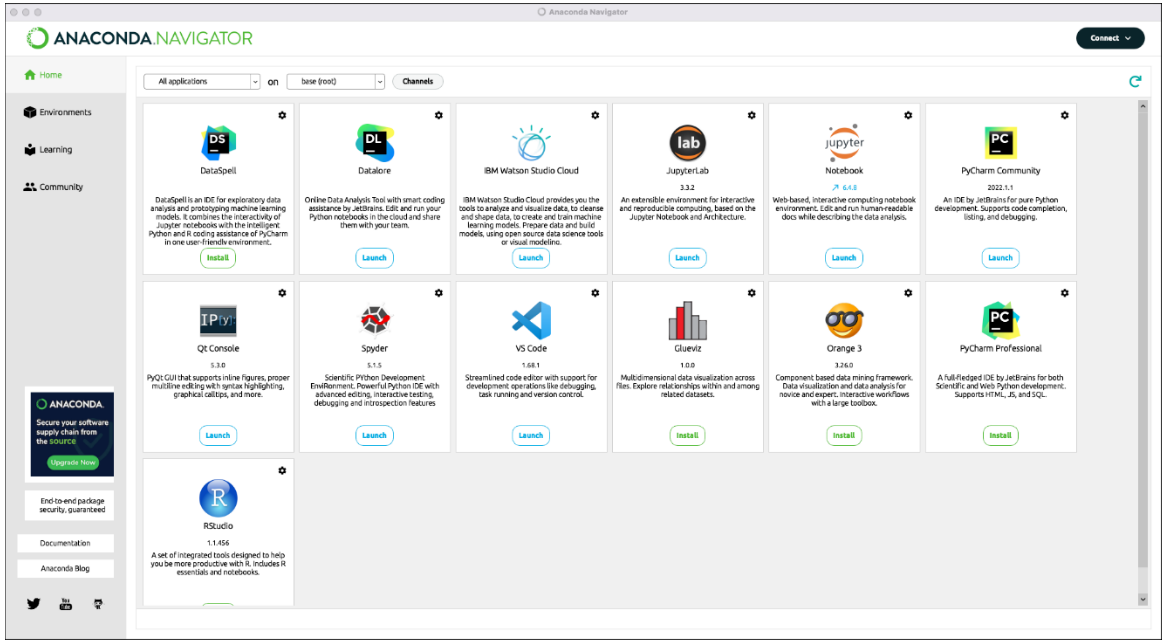
Public

3 videos 26 views Updated 3 days ago

All you need from PyCaret Python library to automate your ML workflow.
Codes are available on my GitHub account: https://github.com/PJalgotrader/platforms-and-tools

Install through ANACONDA

- Anaconda is a distribution of the Python and R programming languages for scientific computing, that aims to simplify package management with conda environments.
- Anaconda offers the easiest way to perform data science and machine learning on a single machine.
- Install Anaconda @ https://www.anaconda.com/



DataSpell
DataSpell is an IDE for exploratory data analysis and prototyping machine learning models. It combines the interactivity of Jupyter notebooks with the intelligent Python and R coding assistance of PyCharm in one user-friendly environment.

Datalore
Online Data Analysis Tool with smart coding assistance by JetBrains. Edit and run your Python notebooks in the cloud and share them with your team.

IBM Watson Studio Cloud
IBM Watson Studio Cloud provides you the tools to analyze and visualize data, to cleanse and shape data, to create and train machine learning models. Prepare data and build models, using open source data science tools or visual modeling.

JupyterLab
3.3.2
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook
↗ 6.4.8
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

PyCharm Community
2022.1.1
An IDE by JetBrains for pure Python development. Supports code completion, listing, and debugging.

Qt Console
5.3.0
PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Spyder
5.1.5
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

VS Code
1.68.1
Streamlined code editor with support for development operations like debugging, task running and version control.

Glueviz
1.0.0
Multidimensional data visualization across files. Explore relationships within and among related datasets.

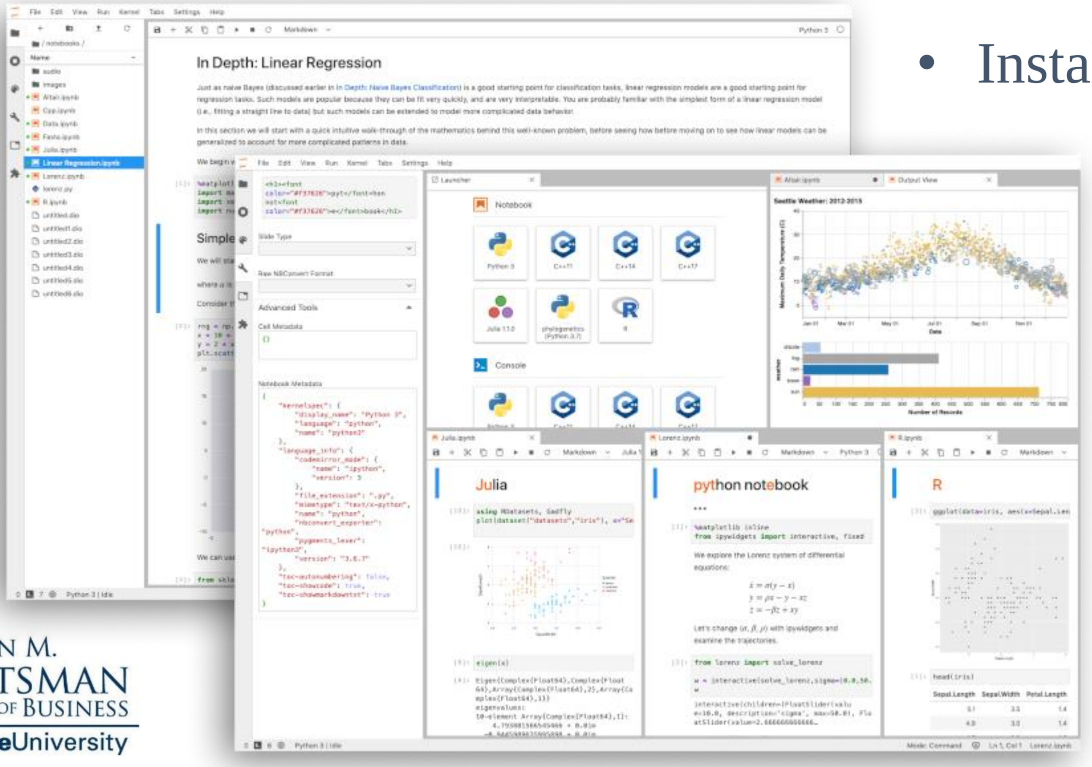
Orange 3
3.26.0
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

PyCharm Professional
A full-fledged IDE by JetBrains for both Scientific and Web Python development. Supports HTML, JS, and SQL.

RStudio
1.1.456
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

JupyterLab

- JupyterLab is the latest web-based interactive development environment for notebooks, code, and data
- Jupyter's name is a reference to the three core programming languages supported by Jupyter, which are Julia, Python and R



In Depth: Linear Regression

Just as naive Bayes (discussed earlier in In Depth: Naive Bayes Classification) is a good starting point for classification tasks, linear regression models are a good starting point for regression tasks. Such models are popular because they can be fit very quickly, and are very interpretable. You are probably familiar with the simplest form of a linear regression model (i.e., fitting a straight line to data) but such models can be extended to model more complicated data behavior.

In this section we will start with a quick intuitive walk-through of the mathematics behind this well-known problem, before seeing how before moving on to see how linear models can be generalized to account for more complicated patterns in data.

```
<h1><font color="#f37626">pyt</font>hon
not<font color="#f37626">e</font>book</h1>
```

```
{
    "kernelspec": {
        "display_name": "Python 3",
        "language": "python",
        "name": "python3"
    },
    "language_info": {
        "codemirror_mode": {
            "name": "ipython",
            "version": 3
        },
        "file_extension": ".py",
        "mimetype": "text/x-python",
        "name": "python",
        "nbconvert_exporter":
"python",
        "pygments_lexer":
"ipython3",
        "version": "3.6.7"
    },
    "toc-autonumbering": false,
    "toc-showcode": true,
    "toc-showmarkdowntxt": true
}
```

Julia

```
using RDatasets, Gadfly
plot(dataset("datasets","iris"), x="Se
```

```
eigen(A)
```

```
Eigen{Complex{Float64},Complex{Float
64},Array{Complex{Float64},2},Array{Co
mplex{Float64},1}}
eigenvalues:
10-element Array{Complex{Float64},1}:
   4.793881566549466 + 0.0im
```

python notebook

```
%matplotlib inline
from ipywidgets import interactive, fixed
```

We explore the Lorenz system of differential equations:

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = \rho x - y - xz$$
$$\dot{z} = -\beta z + xy$$

Let's change (σ, β, ρ) with ipywidgets and examine the trajectories.

```
from lorenz import solve_lorenz
w = interactive(solve_lorenz,sigma=(0.0,50.
w
```

```
interactive(children=(FloatSlider(valu
e=10.0, description='sigma', max=50.0), Flo
atSlider(value=2.666666666666…
```

R

```
ggplot(data=iris, aes(x=Sepal.Len
```

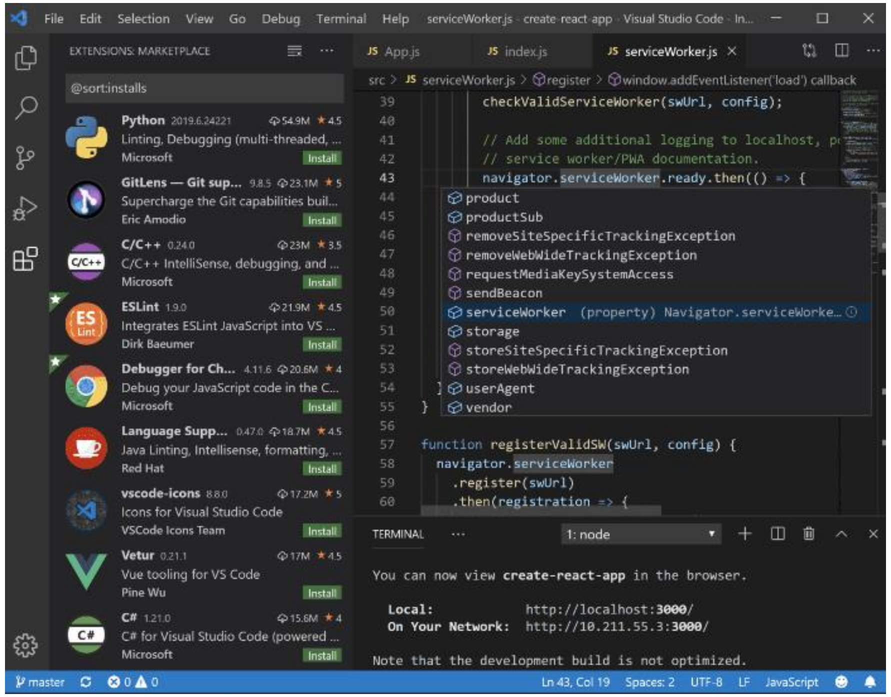
```
head(iris)
```

Sepal.Length	Sepal.Width	Petal.Length
5.1	3.5	1.4
4.9	3.0	1.4

- Install JupyterLab @ http://jupyter.org/install

VS Code

- VS Code is one of the most popular source code editors
- Features include support for debugging, syntax highlighting, intelligent code completion, code refactoring, and embedded Git.
- Install VS code @ https://code.visualstudio.com/


File
Edit
Selection
View
Go
Debug
Terminal
Help
serviceWorker.js - create-react-app - Visual Studio Code - In...
EXTENSIONS: MARKETPLACE
@sort:installs
App.js
index.js
serviceWorker.js
src > serviceWorker.js > register > window.addEventListener('load') callback
product
productSub
removeSiteSpecificTrackingException
removeWebWideTrackingException
requestMediaKeySystemAccess
sendBeacon
serviceWorker (property) Navigator.serviceWorke...
storage
storeSiteSpecificTrackingException
storeWebWideTrackingException
userAgent
vendor
TERMINAL
1: node
master
0 0
Ln 43, Col 19
Spaces: 2
UTF-8
LF
JavaScript

Python 2019.6.24221 54.9M ★ 4.5
Linting, Debugging (multi-threaded, ...
Microsoft Install

GitLens — Git sup... 9.8.5 23.1M ★ 5
Supercharge the Git capabilities buil...
Eric Amodio Install

C/C++ 0.24.0 23M ★ 3.5
C/C++ IntelliSense, debugging, and ...
Microsoft Install

ESLint 1.9.0 21.9M ★ 4.5
Integrates ESLint JavaScript into VS ...
Dirk Baeumer Install

Debugger for Ch... 4.11.6 20.6M ★ 4
Debug your JavaScript code in the C...
Microsoft Install

Language Supp... 0.47.0 18.7M ★ 4.5
Java Linting, Intellisense, formatting, ...
Red Hat Install

vscode-icons 8.8.0 17.2M ★ 5
Icons for Visual Studio Code
VSCode Icons Team Install

Vetur 0.21.1 17M ★ 4.5
Vue tooling for VS Code
Pine Wu Install

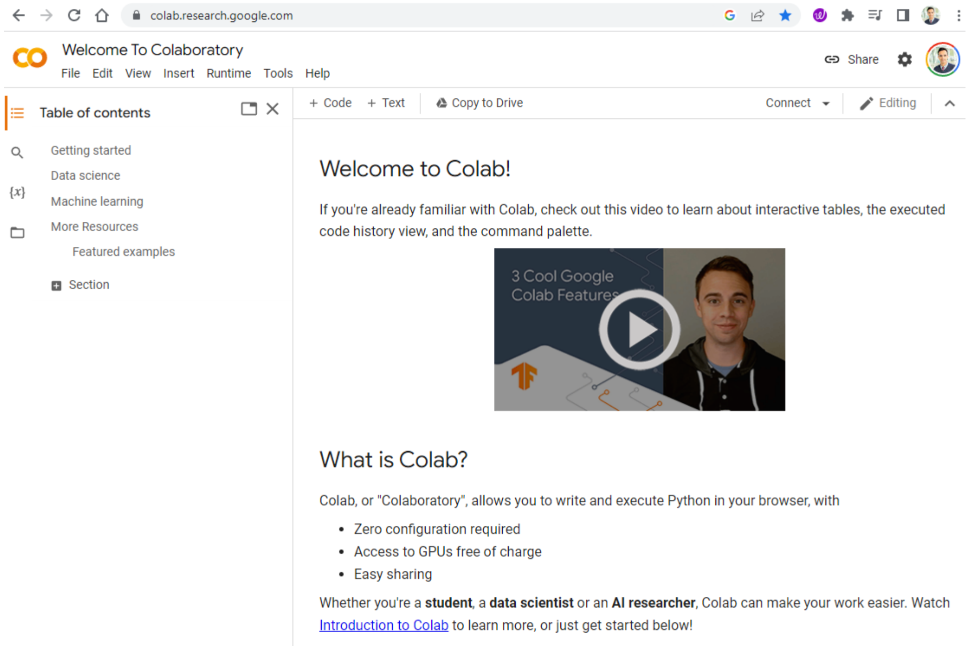
C# 1.21.0 15.6M ★ 4
C# for Visual Studio Code (powered ...
Microsoft Install

```
      checkValidServiceWorker(swUrl, config);

      // Add some additional logging to localhost, p
      // service worker/PWA documentation.
      navigator.serviceWorker.ready.then(() => {
    }
}

function registerValidSW(swUrl, config) {
  navigator.serviceWorker
    .register(swUrl)
    .then(registration => {
```

```
You can now view create-react-app in the browser.

  Local:            http://localhost:3000/
  On Your Network:  http://10.211.55.3:3000/

Note that the development build is not optimized.
```

Google colab

- Colab is a free hosted Jupyter notebook-style environment that runs entirely in the cloud and requires no setup to use. It also provides access to machine learning libraries and computing resources including GPU.
- Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. https://colab.research.google.com/



Welcome to Colab!

If you're already familiar with Colab, check out this video to learn about interactive tables, the executed code history view, and the command palette.

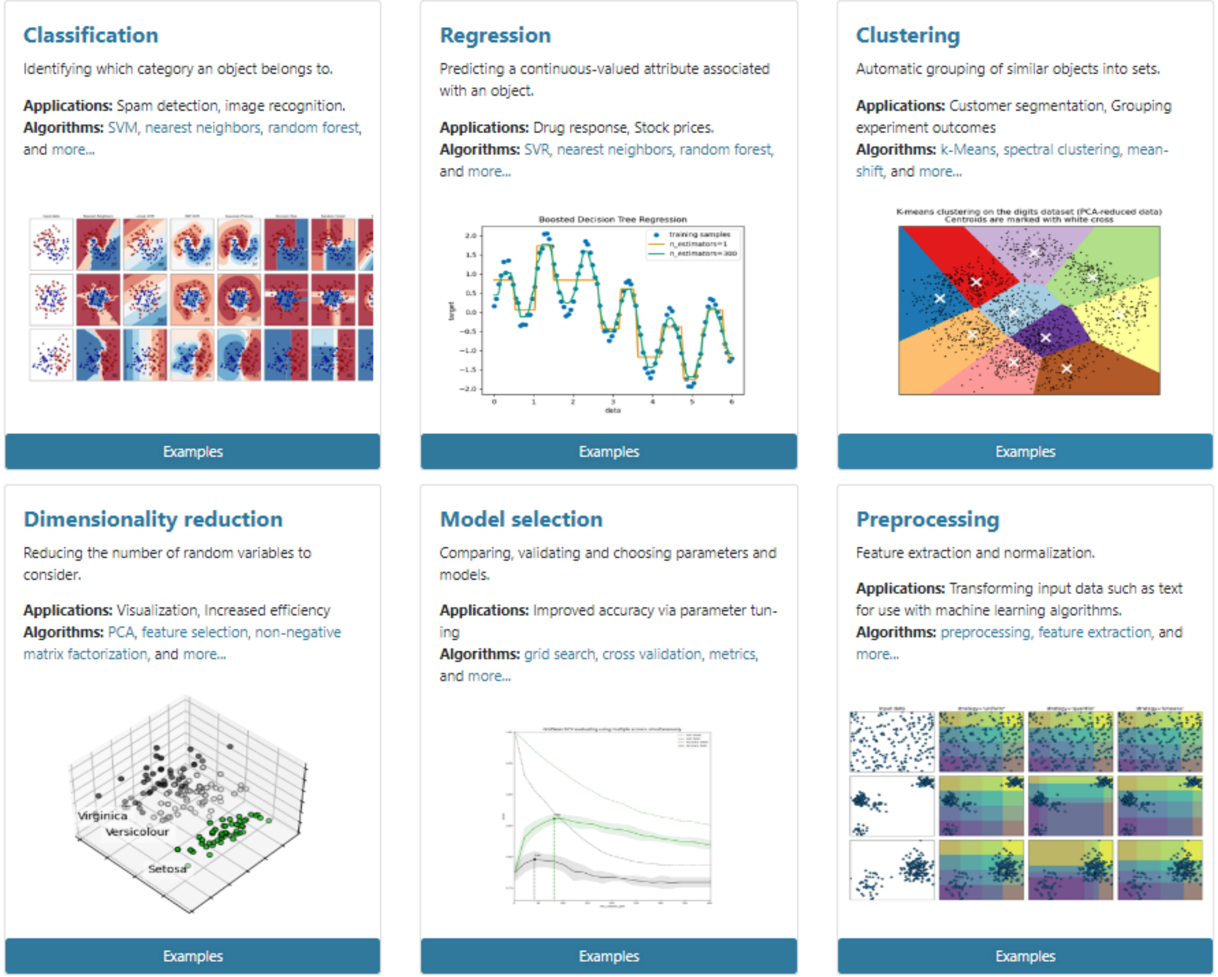
What is Colab?

Colab, or "Colaboratory", allows you to write and execute Python in your browser, with

- Zero configuration required
- Access to GPUs free of charge
- Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch Introduction to Colab to learn more, or just get started below!

scikit learn



Classification

Identifying which category an object belongs to.

Applications: Spam detection, image recognition.
Algorithms: SVM, nearest neighbors, random forest, and more...

Examples

Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.
Algorithms: SVR, nearest neighbors, random forest, and more...

Examples

Clustering

Automatic grouping of similar objects into sets.

Applications: Customer segmentation, Grouping experiment outcomes
Algorithms: k-Means, spectral clustering, mean-shift, and more...

Examples

Dimensionality reduction

Reducing the number of random variables to consider.

Applications: Visualization, Increased efficiency
Algorithms: PCA, feature selection, non-negative matrix factorization, and more...

Examples

Model selection

Comparing, validating and choosing parameters and models.

Applications: Improved accuracy via parameter tuning
Algorithms: grid search, cross validation, metrics, and more...

Examples

Preprocessing

Feature extraction and normalization.

Applications: Transforming input data such as text for use with machine learning algorithms.
Algorithms: preprocessing, feature extraction, and more...

Examples

Boosted Decision Tree Regression
training samples
n_estimators=1
n_estimators=300
target
data
K-means clustering on the digits dataset (PCA-reduced data)
Centroids are marked with white cross
Virginica
Versicolour
Setosa

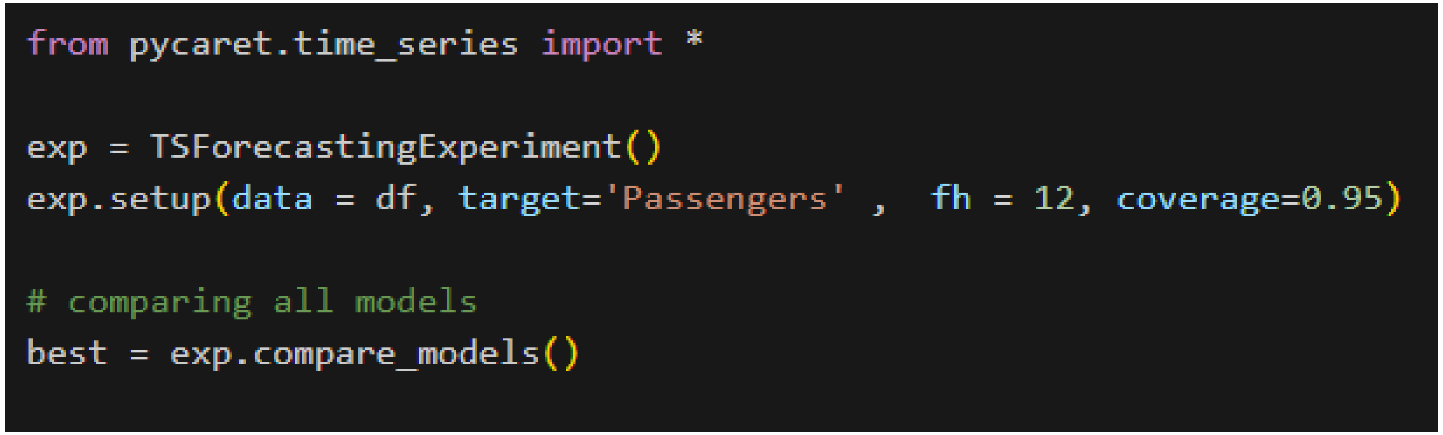
- Scikit-learn is an open-sourced Python library and includes a variety of unsupervised and supervised learning techniques.
- It is based on technologies and libraries like Matplotlib, Pandas and NumPy and helps simplify the coding task.
- Install Scikit-learn @ https://scikit-learn.org/stable/install.html

Jon M. Huntsman School of Business
UtahStateUniversity

Pedram Jahangiry

PYCARET

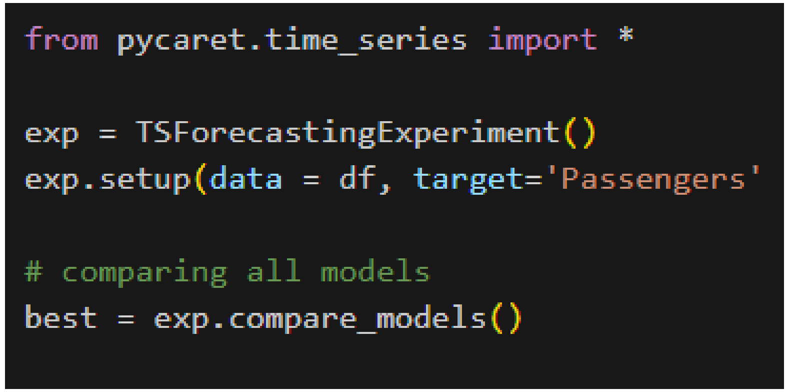
- PyCaret is an open-source, low-code machine/statistical learning library in Python that automates machine learning workflows.
- PyCaret is essentially a Python wrapper around several machine/statistical learning libraries and frameworks
- Install PyCaret @ https://pycaret.gitbook.io/docs/get-started/installation



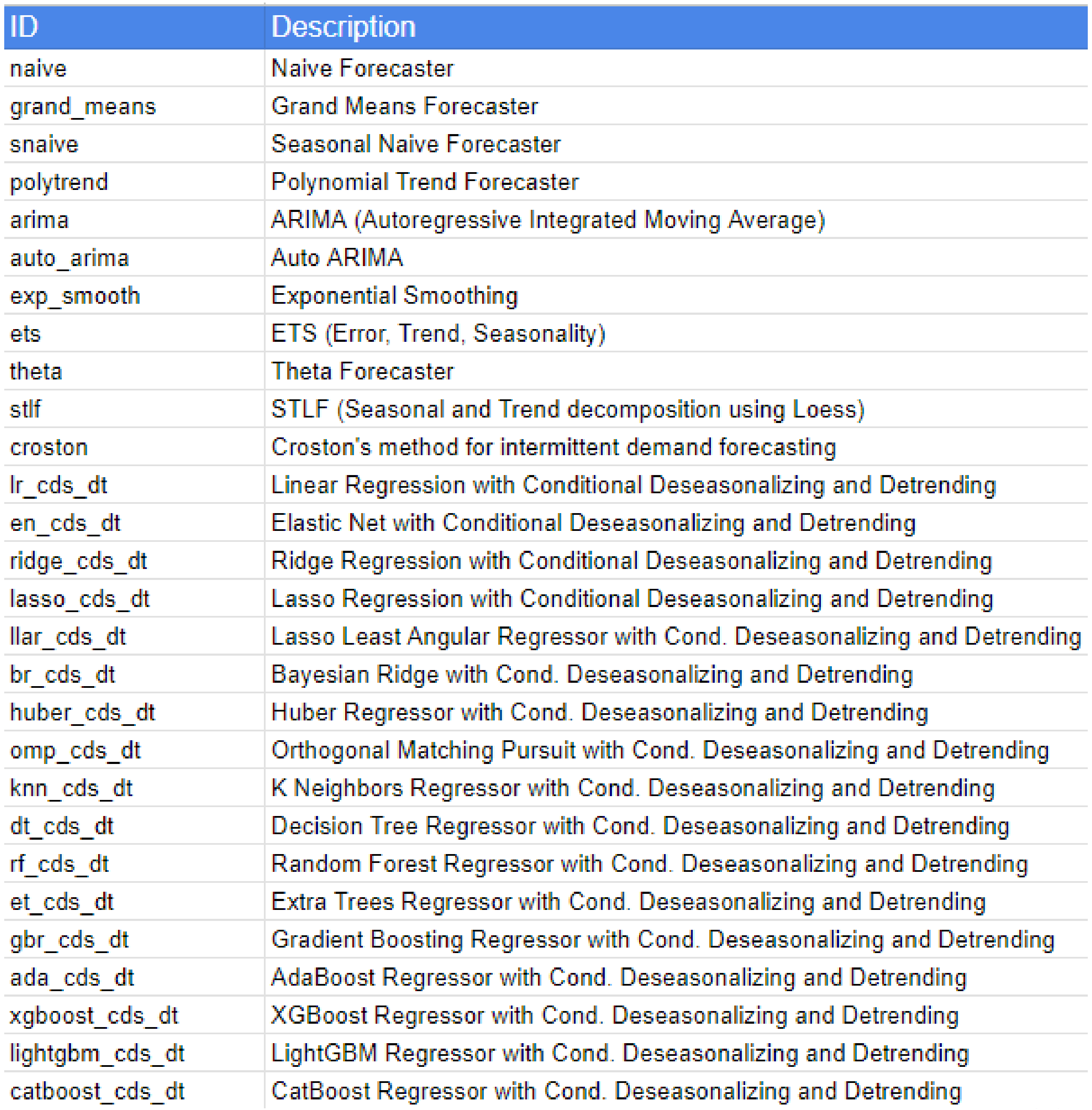
```python
from pycaret.time_series import *

exp = TSForecastingExperiment()
exp.setup(data = df, target='Passengers' ,  fh = 12, coverage=0.95)

# comparing all models
best = exp.compare_models()
```

PYCARET

- List of Timeseries models available in PyCaret
- This is the [Full] version.
- `pip install pycaret[full]`



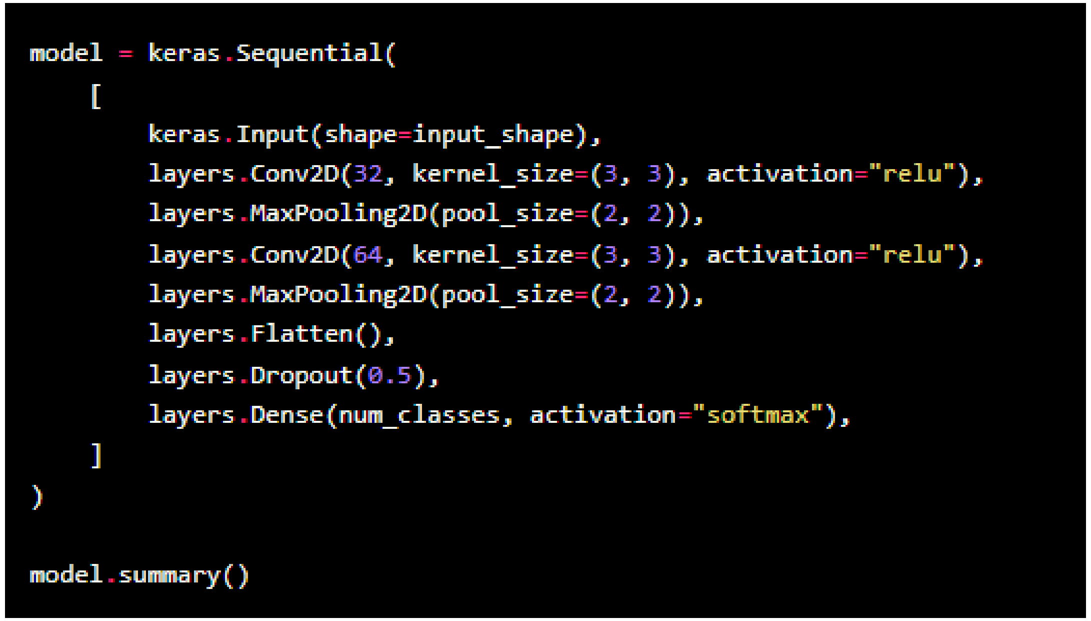
```python
from pycaret.time_series import *

exp = TSForecastingExperiment()
exp.setup(data = df, target='Passengers'

# comparing all models
best = exp.compare_models()
```



ID	Description
naive	Naive Forecaster
grand_means	Grand Means Forecaster
snaive	Seasonal Naive Forecaster
polytrend	Polynomial Trend Forecaster
arima	ARIMA (Autoregressive Integrated Moving Average)
auto_arima	Auto ARIMA
exp_smooth	Exponential Smoothing
ets	ETS (Error, Trend, Seasonality)
theta	Theta Forecaster
stlf	STLF (Seasonal and Trend decomposition using Loess)
croston	Croston's method for intermittent demand forecasting
lr_cds_dt	Linear Regression with Conditional Deseasonalizing and Detrending
en_cds_dt	Elastic Net with Conditional Deseasonalizing and Detrending
ridge_cds_dt	Ridge Regression with Conditional Deseasonalizing and Detrending
lasso_cds_dt	Lasso Regression with Conditional Deseasonalizing and Detrending
llar_cds_dt	Lasso Least Angular Regressor with Cond. Deseasonalizing and Detrending
br_cds_dt	Bayesian Ridge with Cond. Deseasonalizing and Detrending
huber_cds_dt	Huber Regressor with Cond. Deseasonalizing and Detrending
omp_cds_dt	Orthogonal Matching Pursuit with Cond. Deseasonalizing and Detrending
knn_cds_dt	K Neighbors Regressor with Cond. Deseasonalizing and Detrending
dt_cds_dt	Decision Tree Regressor with Cond. Deseasonalizing and Detrending
rf_cds_dt	Random Forest Regressor with Cond. Deseasonalizing and Detrending
et_cds_dt	Extra Trees Regressor with Cond. Deseasonalizing and Detrending
gbr_cds_dt	Gradient Boosting Regressor with Cond. Deseasonalizing and Detrending
ada_cds_dt	AdaBoost Regressor with Cond. Deseasonalizing and Detrending
xgboost_cds_dt	XGBoost Regressor with Cond. Deseasonalizing and Detrending
lightgbm_cds_dt	LightGBM Regressor with Cond. Deseasonalizing and Detrending
catboost_cds_dt	CatBoost Regressor with Cond. Deseasonalizing and Detrending

Keras

- Keras is a high-level, open-source neural network library written in Python. It was developed to make it easier for researchers and developers to build and experiment with deep learning models.
- The Keras API became the official high-level API for TensorFlow 2.0 in 2019. https://keras.io/



```python
model = keras.Sequential(
    [
        keras.Input(shape=input_shape),
        layers.Conv2D(32, kernel_size=(3, 3), activation="relu"),
        layers.MaxPooling2D(pool_size=(2, 2)),
        layers.Conv2D(64, kernel_size=(3, 3), activation="relu"),
        layers.MaxPooling2D(pool_size=(2, 2)),
        layers.Flatten(),
        layers.Dropout(0.5),
        layers.Dense(num_classes, activation="softmax"),
    ]
)

model.summary()
```

Platforms and Packages

Listed below are some Python packages and platforms that will be used in the deep learning and deep forecasting courses.

Google colab

General Python libraries

NumPy

pandas

matplotlib

seaborn

Machine Learning libraries

scikit learn

dmlc XGBoost

LightGBM

PYCARET

SKTIME

PROPHET

Deep Learning libraries

K Keras

TensorFlow

Setting up Deep Learning Environment

	Personal Workstation	Cloud Platforms	Google Colaboratory
Pros	• Full control over hardware and software • Work offline • Fixed cost	• Powerful computing resources • Scalability • Ease of use • Cost-effective: Pay-as-you-go • Collaboration	• Powerful computing resources (GPU, TPU) • Ease of use • Collaboration • No need to set up a local environment
Cons	• Scalability • Maintenance (both hardware and software)	• Expensive for large-scale experiments • Dependency on the provider • Limited control • Internet connection • Security	• Time limit • Hardware limitation • Data storage • Limited control • Internet connection • Security

Kaggle Survey 2022

- Kaggle runs a yearly survey among machine learning and data science professionals worldwide.
- This survey is one of our most reliable sources about the state of the industry!!!

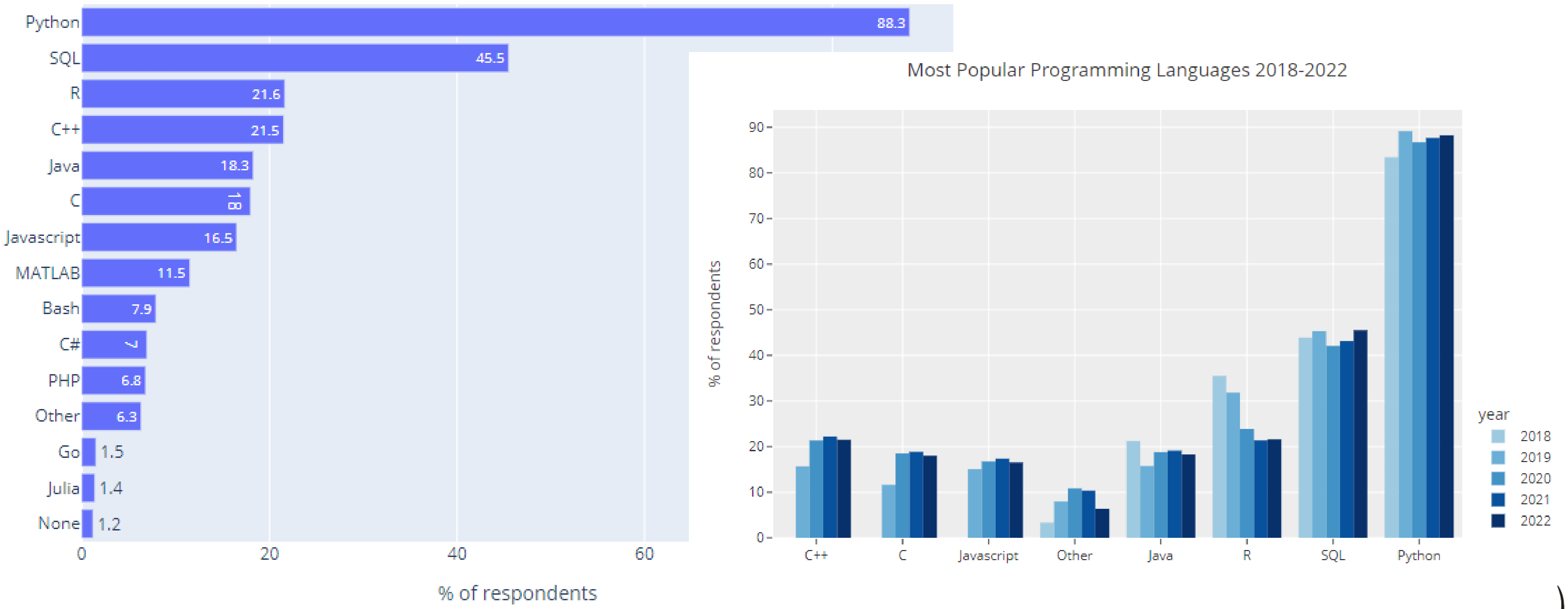
Jon M. Huntsman School of Business Utah State University

Pedram Jahangiry

https://www.kaggle.com/kaggle-survey-2022

Programming Languages

Most Popular Programming Languages in 2022



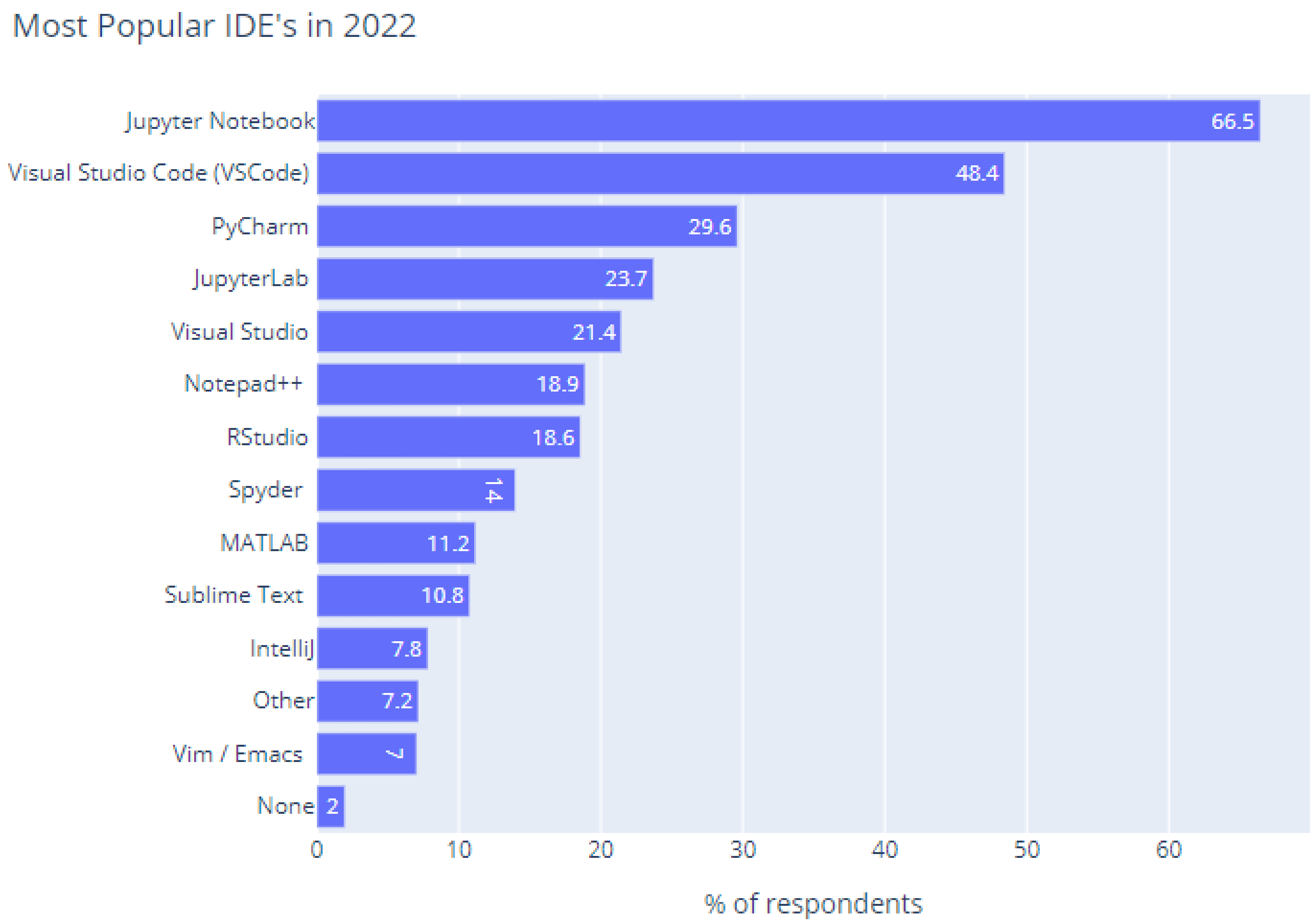
Most Popular Programming Languages 2018-2022

Jon M. Huntsman School of Business
Utah State University

Pedram Jahangiry

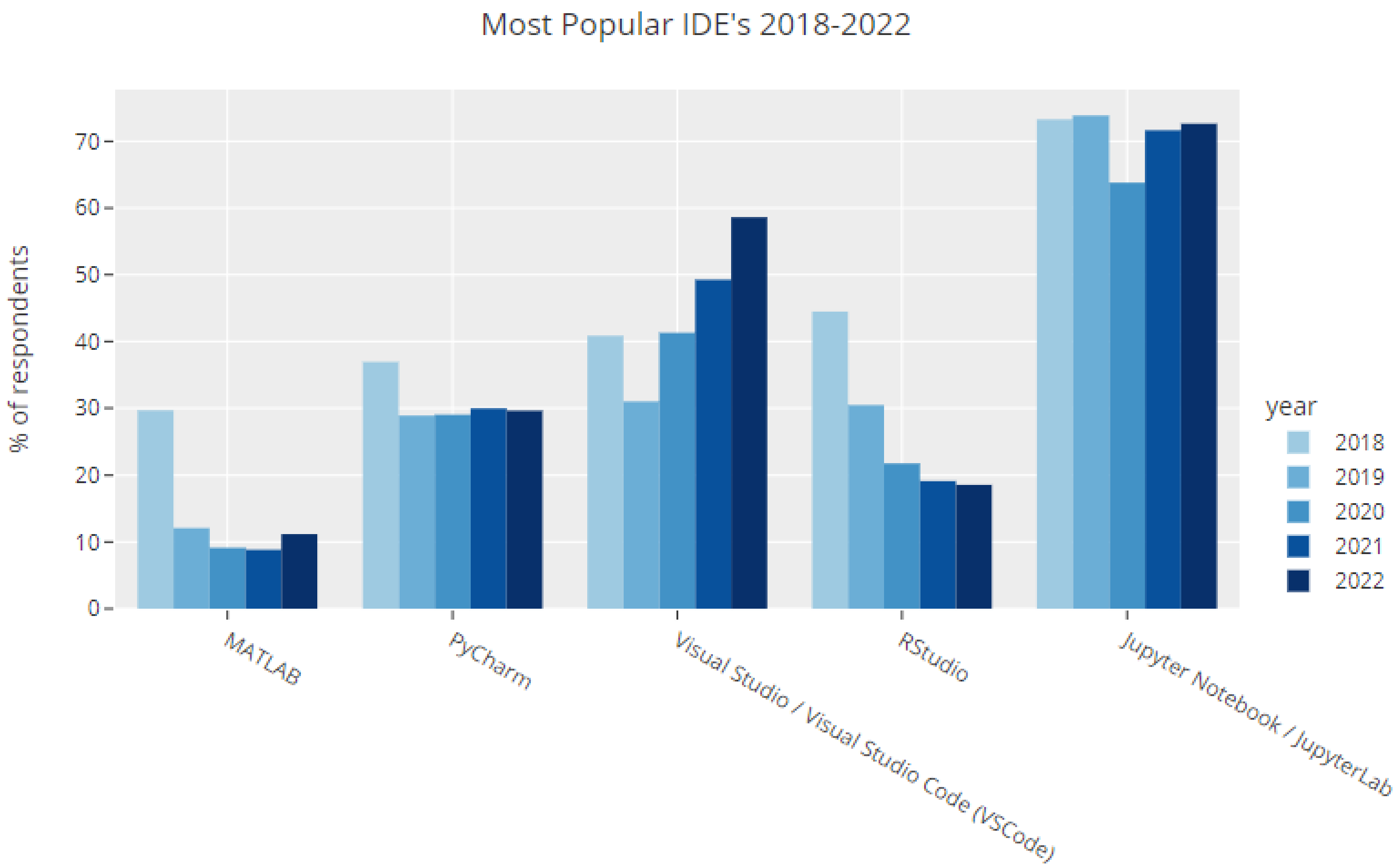
https://www.kaggle.com/kaggle-survey-2022

Popular IDE’s



https://www.kaggle.com/kaggle-survey-2022

Popular IDE's



https://www.kaggle.com/kaggle-survey-2022

Python packages

Package	Metrics	ML	DL	Focus
statsmodels	✓✓✓✓			Statistics, Econometrics
scikit learn	✓	✓✓✓✓		General Machine Learning
K Keras			✓✓✓✓	General Deep Learning
PYCARET	✓✓✓	✓✓✓		Auto ML, Rapid prototyping, Comparison
Darts	✓✓✓	✓	✓✓	Advanced timeseries and forecasting
GLUONTS	✓	✓	✓✓	Large Scale probabilistic Models
NIXTLA	✓✓✓	✓✓✓	✓✓✓	Workflows, SOTA methods

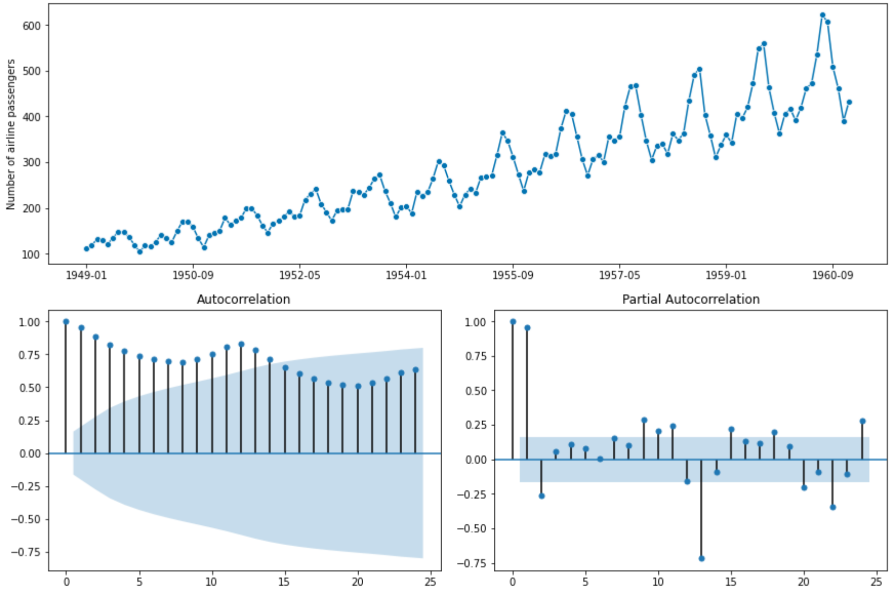
Jon M. Huntsman School of Business
Utah State University

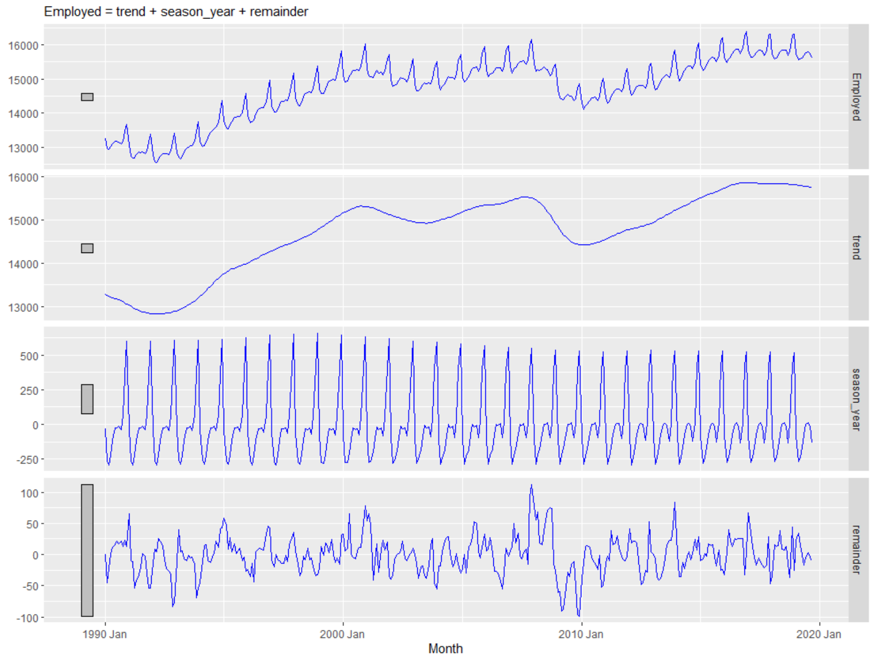
Pedram Jahangiry

Module 2 – Part II

Setting up Deep Forecasting Environment

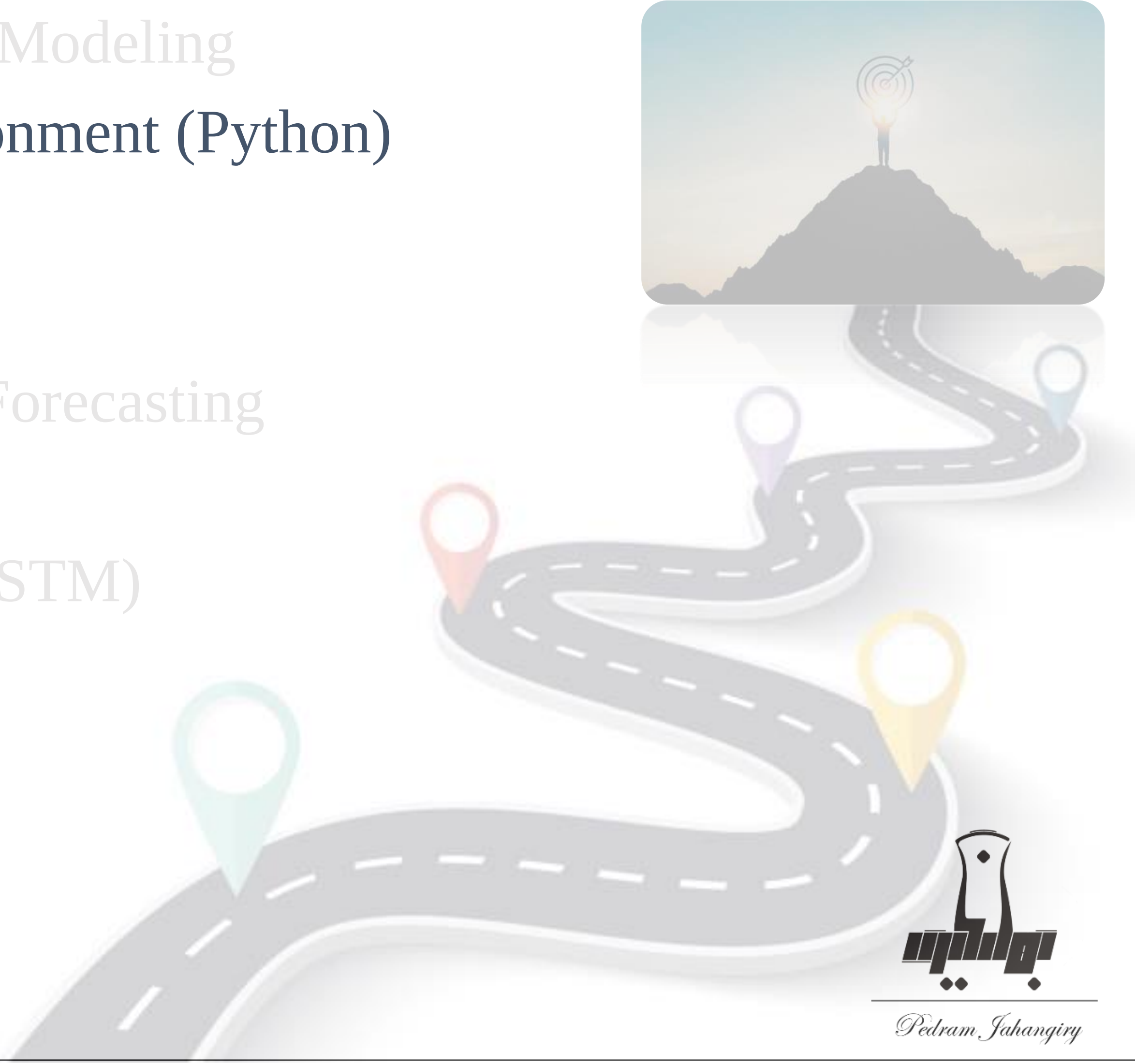
(Time Series Basics in Python)





Road map!

- Module 1- Demystifying Timeseries Data and Modeling
- Module 2- Setting up Deep Forecasting Environment (Python)
- Module 3- Exponential Smoothing
- Module 4- ARIMA models
- Module 5- Machine Learning for Time series Forecasting
- Module 6- Deep Neural Networks
- Module 7- Deep Sequence Modeling (RNN, LSTM)
- Module 8- Prophet and Neural Prophet



Jon M. Huntsman School of Business
UtahStateUniversity

Pedram Jahangiry

Road map!

- ✓ Module 1- Demystifying Timeseries Data and Modeling
- ✓ Module 2- Setting up Deep Forecasting Environment (Python)
- Module 3- Exponential Smoothing
- Module 4- ARIMA models
- Module 5- Machine Learning for Time series Forecasting
- Module 6- Deep Neural Networks
- Module 7- Deep Sequence Modeling (RNN, LSTM)
- Module 8- Prophet and Neural Prophet