Module 2 – Part I

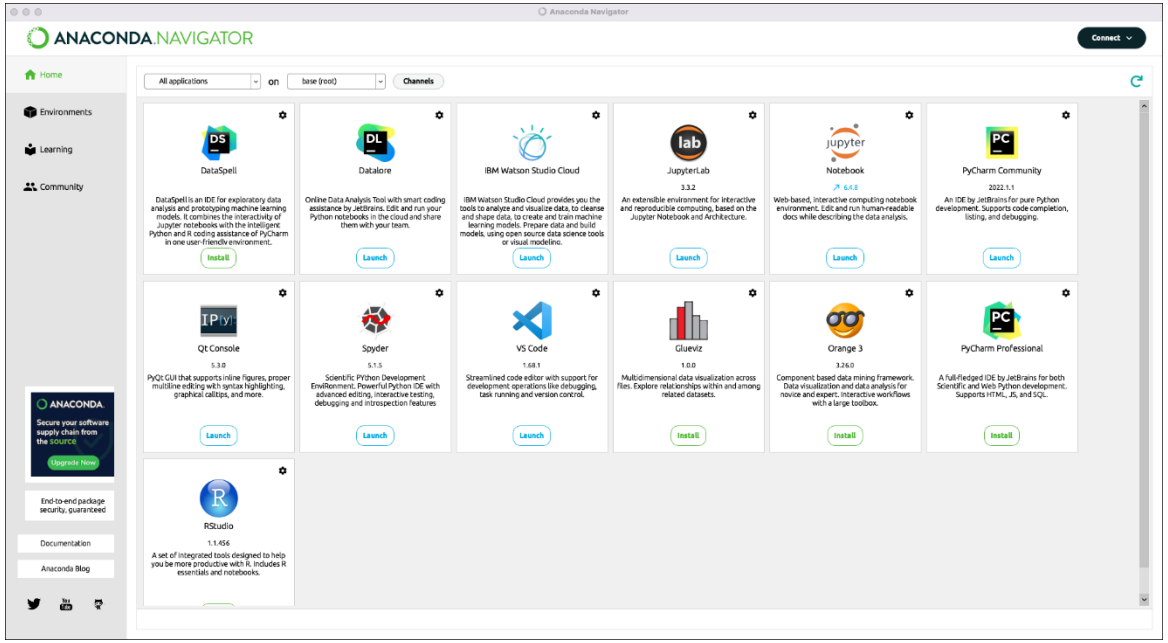
Setting up Deep Forecasting Environment (Python)

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

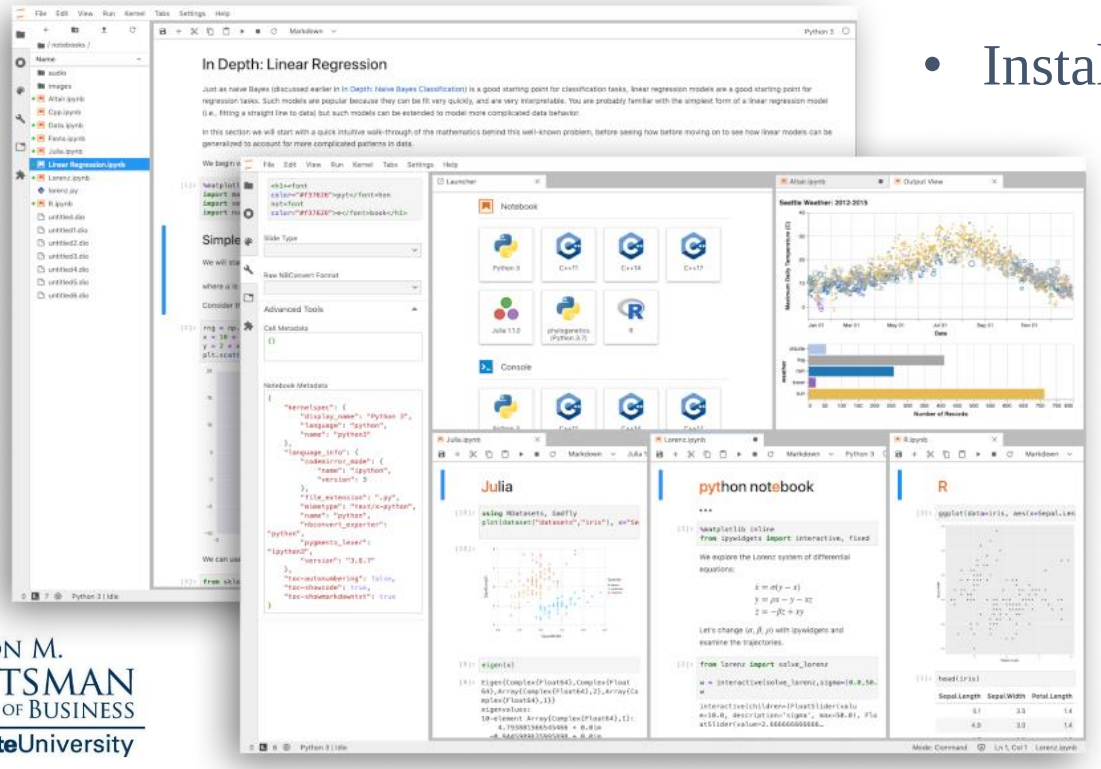
Install through

- 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/



JupyterLab

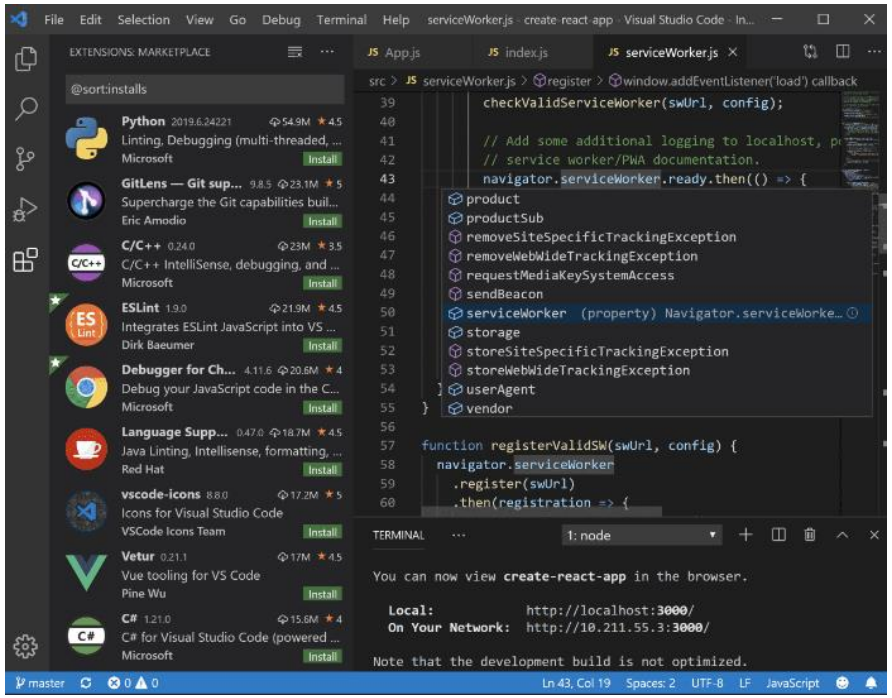
- JupyterLab is the latest web-based interactive development environment for notebooks, code, and data
- Jupyter's name is a reference to the three core programming languages supported by Jupyter, which are Julia, Python and R



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

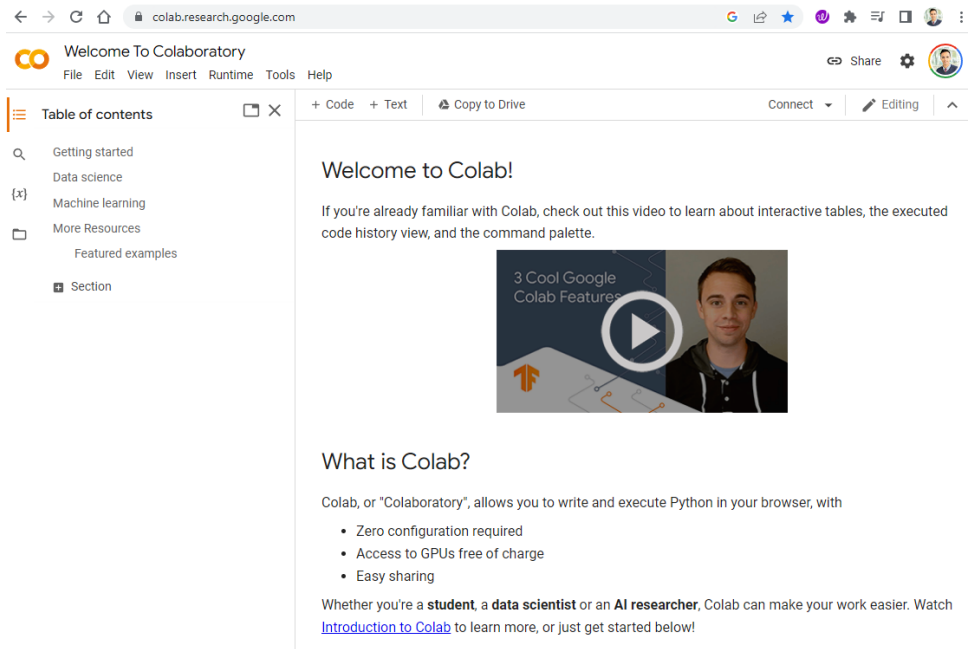
VS Code

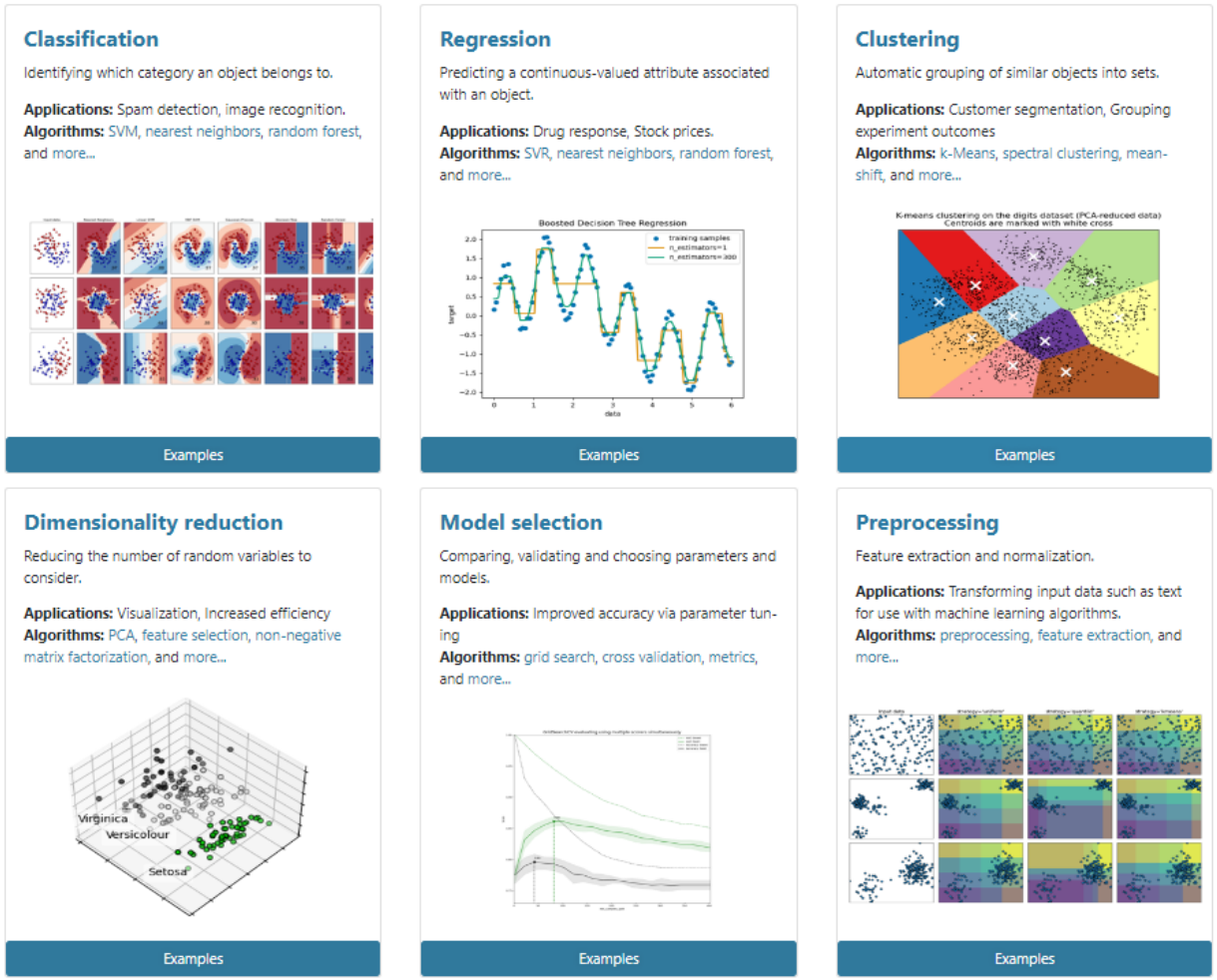
- VS Code is one of the most popular source code editors
- Features include support for debugging, syntax highlighting, intelligent code completion, code refactoring, and embedded Git.
- Install VS code @ https://code.visualstudio.com/



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/





- 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

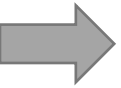
PYCARET

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

```
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]`

```
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

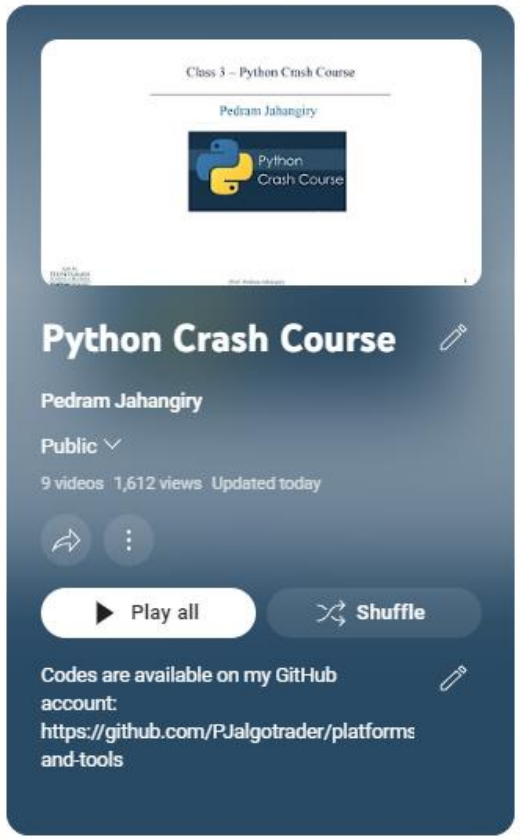
Keras

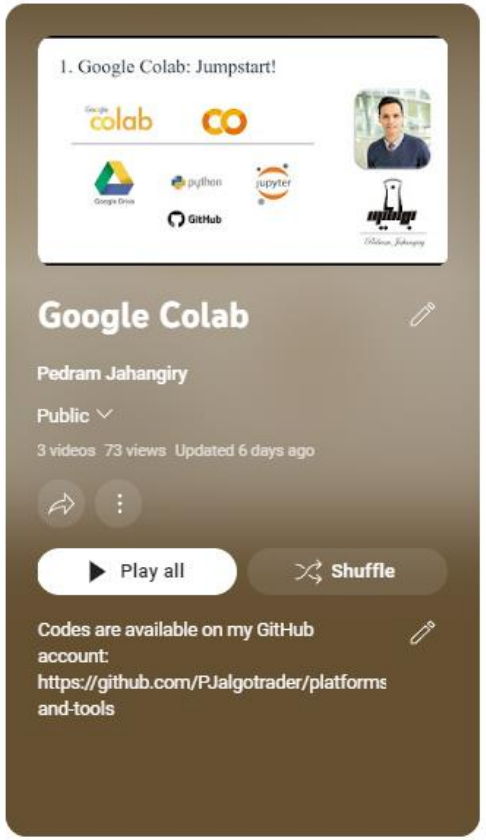
- Keras is a high-level, open-source neural network library written in Python. It was developed to make it easier for researchers and developers to build and experiment with deep learning models.
- The Keras API became the official high-level API for TensorFlow 2.0 in 2019. https://keras.io/

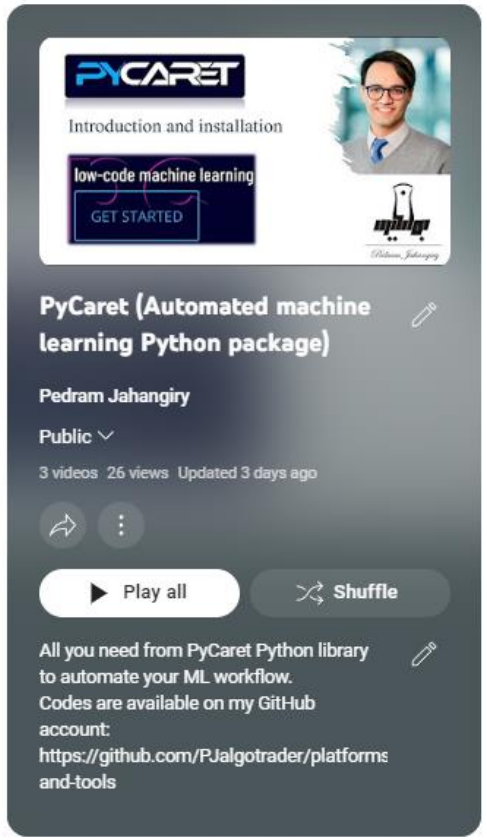
```
model = keras.Sequential(
    [
        keras.Input(shape=input_shape),
        layers.Conv2D(32, kernel_size=(3, 3), activation="relu"),
        layers.MaxPooling2D(pool_size=(2, 2)),
        layers.Conv2D(64, kernel_size=(3, 3), activation="relu"),
        layers.MaxPooling2D(pool_size=(2, 2)),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

model.summary()
```

Available YouTube playlists







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

Machine Learning libraries

Deep Learning libraries

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

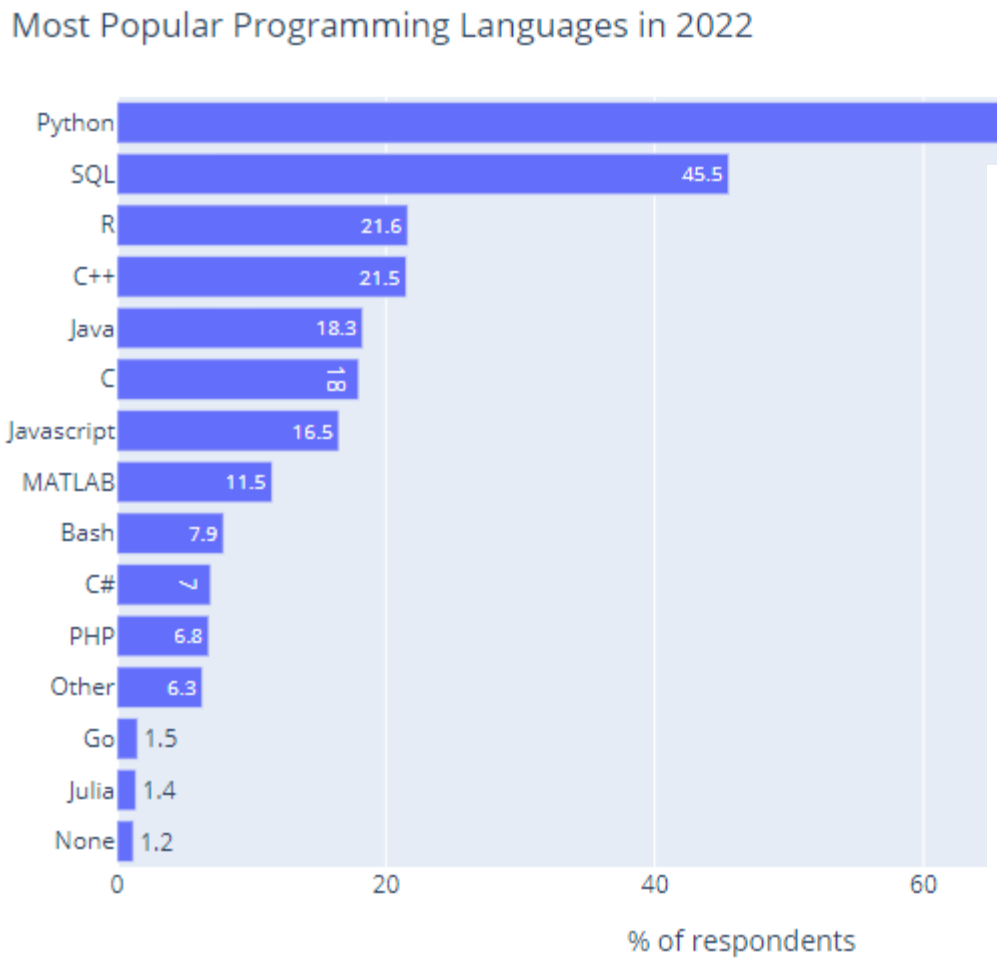
Jon M. Huntsman School of Business
UtahStateUniversity

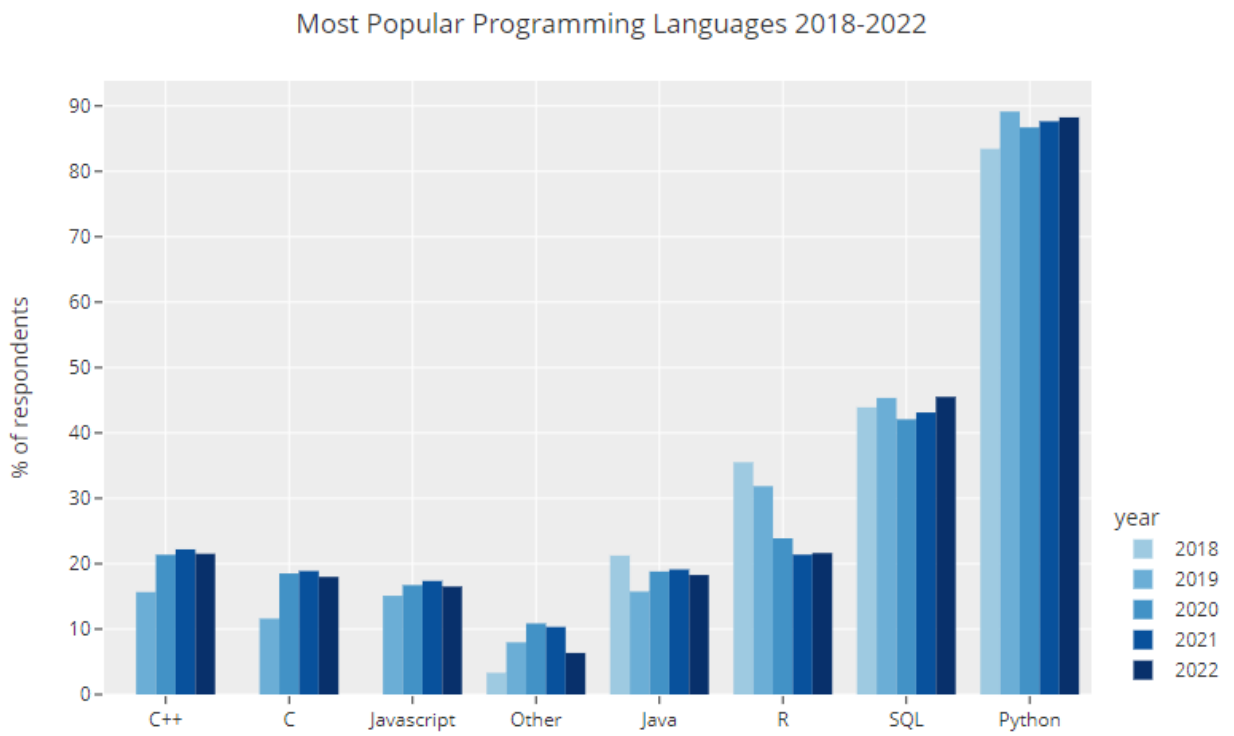
Pedram Jahangiry

Kaggle Survey 2022

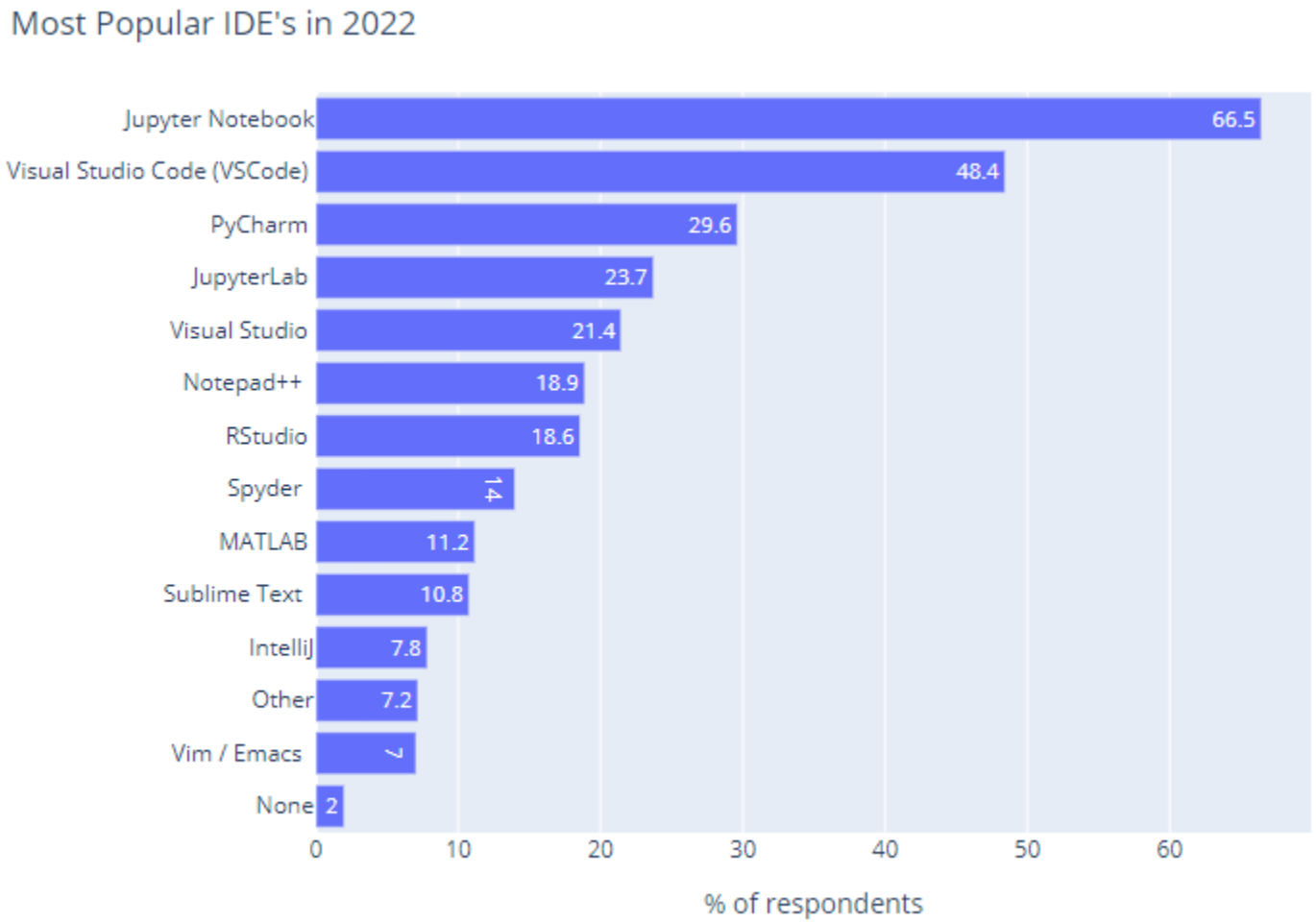
- Kaggle runs a yearly survey among machine learning and data science professionals worldwide.
- This survey is one of our most reliable sources about the state of the industry!!!

Programming Languages

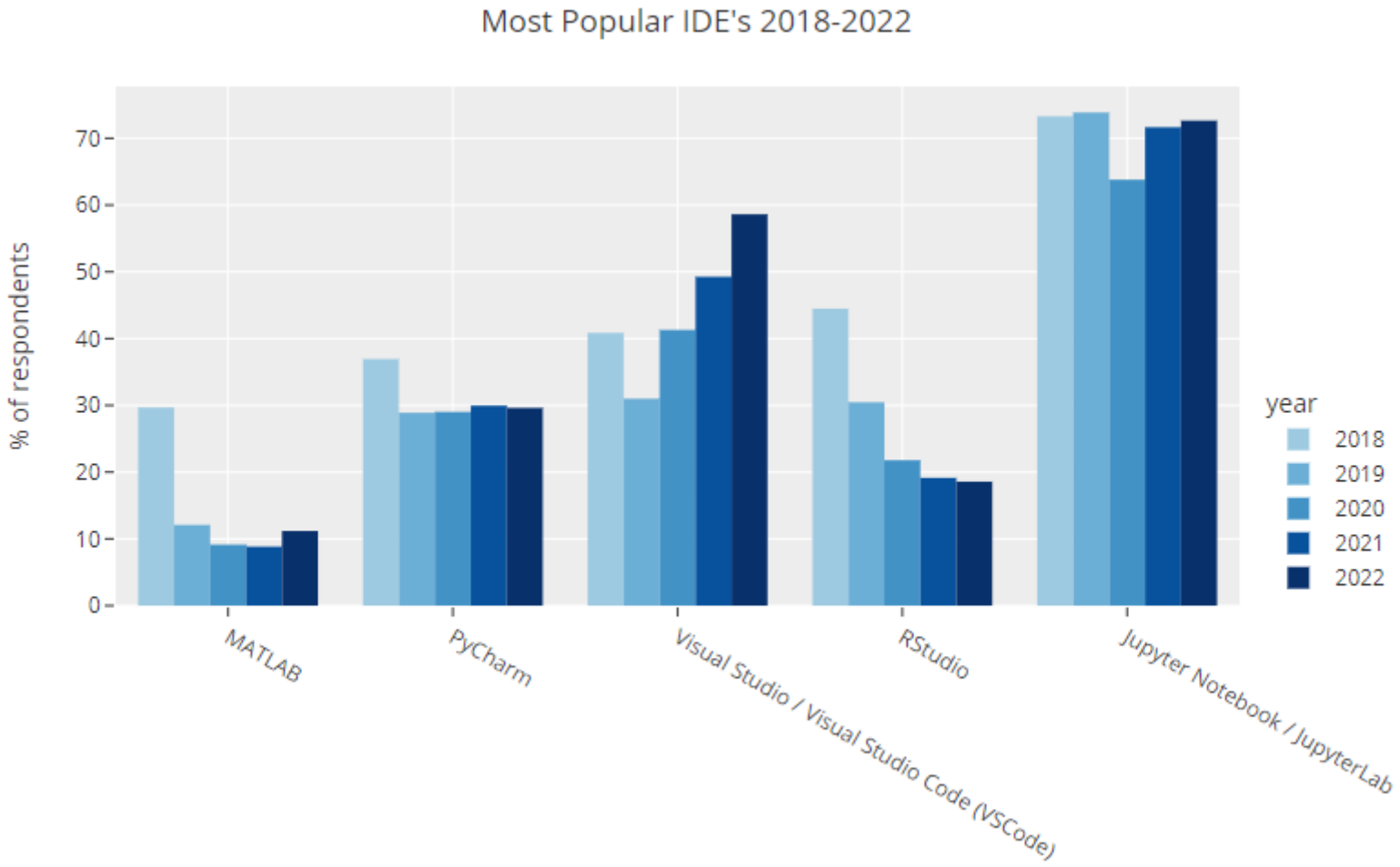




Popular IDE's



Popular IDE's



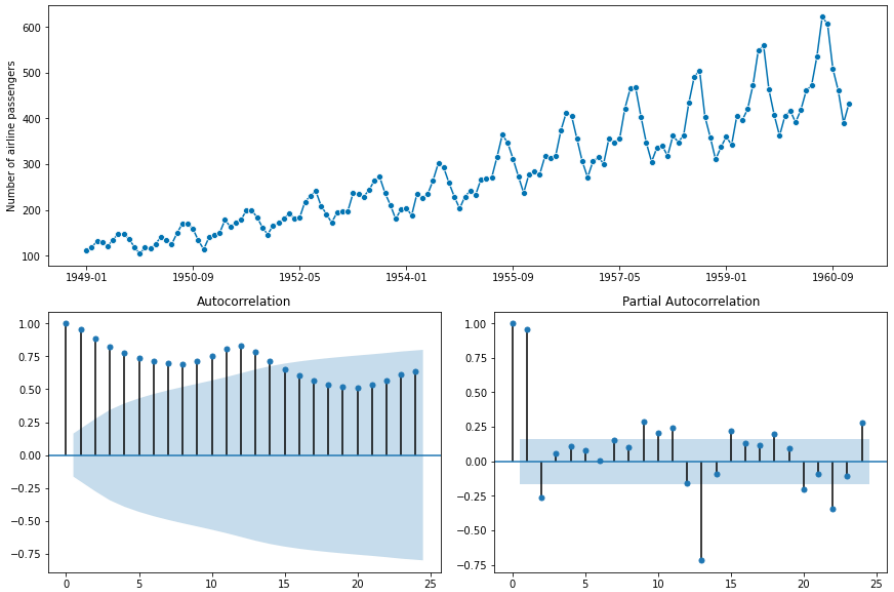
Python packages

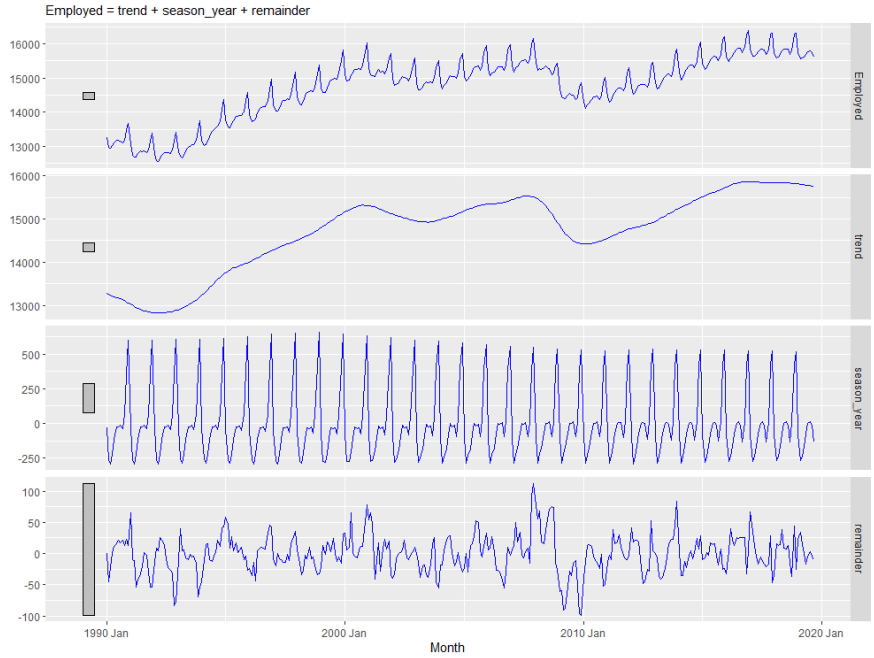
Package	Metrics	ML	DL	Focus
statsmodels	✓✓			Statistics, Econometrics
scikit learn	✓	✓✓		General Machine Learning
Keras			✓✓	General Deep Learning
PYCARET	✓✓	✓✓		Auto ML, Rapid prototyping, Comparison
Darts	✓✓	✓	✓✓	Advanced timeseries and forecasting
GLUONTS	✓	✓	✓✓	Large Scale probabilistic Models
NIXTLA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

Module 2 – Part II

Setting up Deep Forecasting Environment

(Time Series Basics in Python)





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

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