ISEP:

* Master’s degree in mechanical engineering – Mathematics, Advance Statistical, Programming (VBA, Python), linear optimization models.

Experience (6,5 years):

* Freelancer:
  + Collaborating with "Ivy - The Unified Machine Learning Framework" to solve issues on their GitHub repository. (NumPy, TensorFlow, Docker)
  + Engaged in self-directed machine learning projects, seeking opportunities to deepen knowledge and expand expertise in the field.
* Yazaki:
  + Successfully developed and deployed a linear regression model for predicting injection tool costs, resulting in increased efficiency and cost savings in project estimations. This was achieved through the creation of a fully automated Excel file. Kiazen

MY current job experience skills that are transferable to the new job such as:

* Efficient problem-solving.
* Critical thinking.
* Conflict Management and Negotiation Strategies.
* **Estratégias de Gestão de Conflitos e Negociação**
* Attention to detail.
* Project management

I have taken several courses and earned certifications in

* **Machine Learning** Specialization (Pack of 3 courses. All completed on Linkedin).
* **Deep Learning** Specialization by Andrew Ng (Pack of 5 courses all on Linkedin).
* **Spark** and Python for Big Data with **PySpark** (Databricks, AWS EC2)
* **Apache Kafka.**
* University of California, San Diego - Rady School of Management logo
* Introduction to Big Data (**Hadoop**)
* The Ultimate **MySQL** Bootcamp: Go from **SQL** Beginner to Expert.
* **Natural language processing** – Hugging Face course.
* Introduction to **Statistics** from Stanford University.
* Eric Matthes - Python Crash Course\_ A Hands-On (Projects on my GitHub).
* Complete **Python** Bootcamp.
* Microsoft Excel - Advanced Level.

**Python, SQL, Docker, TensorFlow/Keras, Pandas, SciPy, Matplotlib, Scikit-Learn, NumPy, PySpark, Kafka, Github/Gitlab.**

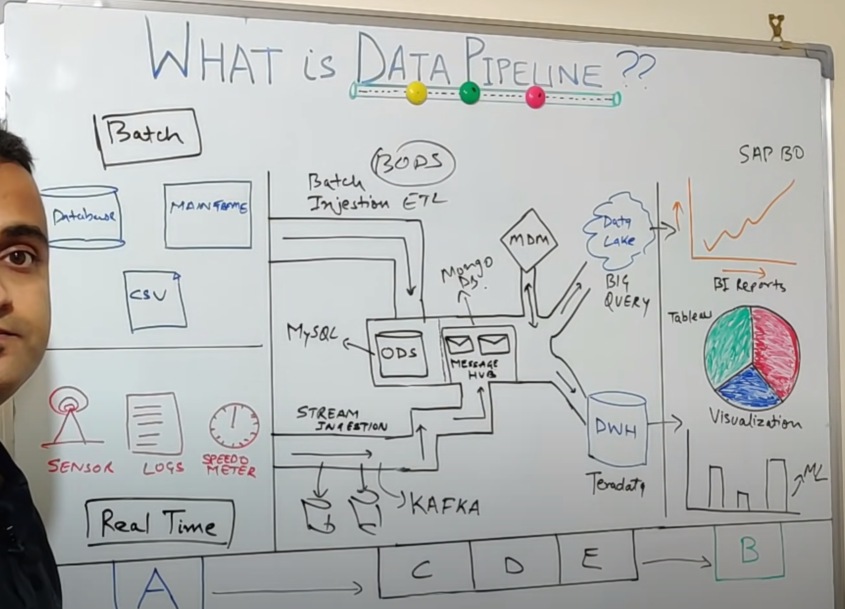
**Docker** – Used to run function and unit test on Ivy open-source repository.

**Pyspark**: Python API for Apache Spark – Used on Databricks platform and on Pycharm for SQL extract, transform and load data. Also, to build and deploy ML models.

\*\*Criar apresentação sobre os meus projetos

**Uma imagem com texto

Descrição gerada automaticamente**

****

**Uma imagem com diagrama

Descrição gerada automaticamente**

**ML model Set up and initialization:**

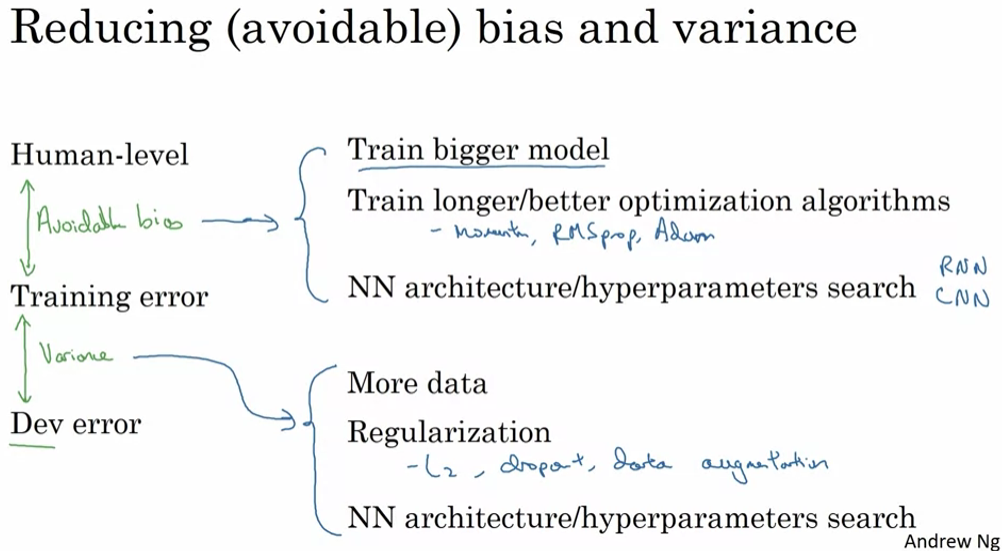
* Train / Dev / Test sets
* Regularization:
  1. L2 regularization.
  2. Dropout regularization.
  3. Data augmentation + Data synthesis.
  4. Normalizing Inputs.

**Dev. Set (Cross-validation):**

* Machine learning engineers, use this data to fine-tune the model hyperparameters.
* The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters.

**ML optimization:**

* Hyperparameter Tuning.
  1. Scale hyperparameters.
  2. Tuning in Practice: Pandas vs. Caviar.
  3. Batch normalization
* Gradient descent
* Deep learning optimization:
  1. Mini-batch Gradient Descent.
  2. Exponentially Weighted Averages.
  3. Bias Correction in Exponentially Weighted Averages.
  4. Gradient Descent with Momentum.
  5. RMSprop.
  6. Adam Optimization Algorithm (Combines the 1, 3 4 and 5)
* Learning Rate Decay

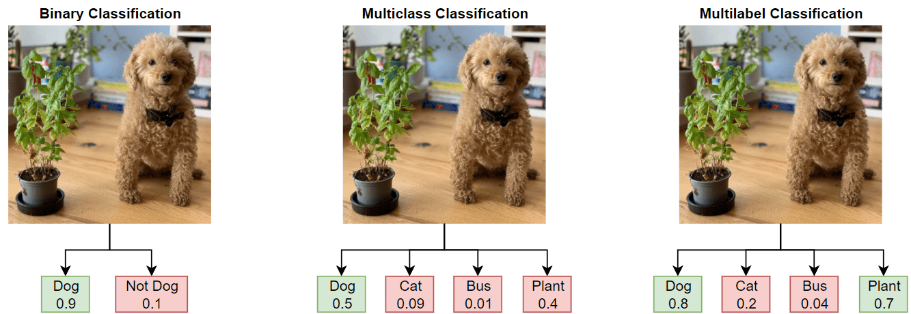


**Model quality and evaluation (performance metrics):**

* **Evaluation**:
  + Bias and Variance - evaluation metrics to plan a strategy for ML optimization.
  + Regression metrics: (Cost functions)
    - Mean Absolute Error (MAE),
    - Mean Squared Error (MSE),
    - Root Mean Squared Error (RMSE),
    - R² (R-Squared).
    - Root Mean Squared Logarithmic Error (RMSLE)
  + Classification metrics:
    - Accuracy.
    - Confusion Matrix (not a metric but fundamental to others):
      * Precision – If the positive observations are true positives.
      * Recall – If there are true positives being misclassified.
    - F1-score.
    - Area Under the ROC Curve (AUC – ROC).

**ML course:** Tensorflow/Keras, PyTorch, Scikit-Learn, NumPy. Matplotlib

Supervised learning:

* Linear regression. (Cost function, gradient descent)
* Logistic regression.
* Neural networks (Deep learning) – Ml libraries like: PyTorch, Tensorflow, Keras.
  + Linear regression.
  + logistic regression:
    - Binary.
    - Multiclass.
    - Multilabel.
  + Convolutional layer (CNN):
    - Image classification:
      * LetNet
      * AlexNet
      * VGG-16
      * ResNets
      * MobileNetV2
    - Image classification with Object Localization or Semantic Segmentation:
      * YOLO Algorithm.
      * R-CNN.
      * U-Net (among other features, it uses Semantic Segmentation).
* Decision Trees / Tree ensemble:
  + XGboost

Unsupervised learning:

* Clustering: K-means, anomaly detection (Scikit-learn).
* Recommendations: Collaborative filtering algorithm and Content-based filtering (PyTorch, Tensorflow).
* Reinforcement learning.

**REST API’s:**

**Uma imagem com texto, pessoa, quadro-preto

Descrição gerada automaticamente**

**Uma imagem com texto

Descrição gerada automaticamente**

**Git:**

****

**Docker:**

It is a containerization platform that allows developers to package code into various deployable units called containers. Docker Engine is the software that oversees the hosting of the containers.

Docker version, Docker search, Docker pull, Docker run, Docker ps, Docker stop, Docker restart, Docker kill, Docker exec, Docker login, Docker commit, Docker push, Docker network, Docker history, Docker rmi, Docker ps -a, Docker copy, Docker logs, Docker volume, Docker logout.

**Use cases:**

Neural networks (Deep learning): Ml algorithm libraries like: PyTorch, Tensorflow, Keras.

* Computer vision:
  + Image classification
    - Uses binary, multi-label and multi-class classification to classify and distinguish different objects in static or dynamic images for purposes like facial recognition and self-driving cars.
    - Object detection that uses the image classification search for certain patterns that can be used in Medical imaging and Manufacturing control (visual inspection/quality control).
    - Object tracking.
* NLP - Natural language processing
  + Virtual agents and chat-bots (Siri, Alexa, google assistant)
  + Speech recognition.
  + Spam detection
  + Machine translation

Unsupervised Learning

* Clustering:
  + Search Result Clustering
  + Market and Customer segmentation
* Anomaly detection:
  + Fraud detection.
  + Manufacturing maintenance and quality control.
  + Monitoring computers in data centres.
* Recommendations systems
  + e-Commerce
  + Retail
  + Media
  + Banking
* Reinforcement learning:
  + Self-driving cars
  + Industry automation using robots.
  + NLP (Natural Language Processing)
  + Gaming industry.

Uma imagem com texto

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto, mesa

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente