**Things to know must before diving into Machine Learning**

Learning **Machine Learning (ML)** requires a strong foundation in a few essential areas before diving into algorithms and models. Here’s a roadmap:

**1. Mathematics Foundations**

ML relies heavily on mathematics. Focus on the following topics:

**Linear Algebra:**

* Vectors, matrices, and operations.
* Eigenvalues, eigenvectors, and matrix decomposition (important for PCA, embeddings).

**Statistics and Probability:**

* Descriptive stats (mean, median, variance, standard deviation).
* Probability basics: Bayes' theorem, conditional probability.
* Probability distributions (Gaussian, Bernoulli, Poisson, etc.).

**Calculus:**

* Basics of derivatives and integrals.
* Partial derivatives and gradient (for optimization in ML algorithms).

**Discrete Mathematics (optional):**

* Useful for understanding algorithms and data structures.

**2. Programming Skills**

You need strong coding skills, especially in a language popular for ML:

**Recommended Language:**

* **Python**: Most widely used, with powerful libraries like NumPy, Pandas, Scikit-learn, TensorFlow, PyTorch, etc.

**Key Concepts to Learn:**

* Data structures (lists, dictionaries, sets).
* Loops, conditionals, functions, and object-oriented programming.
* Basic libraries: NumPy (arrays), Pandas (dataframes), Matplotlib/Seaborn (visualization).

**3. Data Handling and Preprocessing**

Data is the backbone of ML. You must know:

* How to collect, clean, and preprocess data.
* Handling missing values, outliers, and feature scaling (normalization, standardization).
* Exploratory Data Analysis (EDA) with tools like Pandas and Seaborn.

**4. Basic Understanding of Algorithms**

Learn how algorithms work and why they're used. Start with:

* Sorting and searching algorithms.
* Complexity (Big-O notation).
* Basics of optimization and gradient descent.

**5. Familiarity with Linear Models and Basic ML Concepts**

Before diving deep, understand:

* Regression (linear, logistic).
* Overfitting and underfitting.
* Cross-validation.
* Supervised vs. unsupervised learning.

**6. Tools and Platforms**

* **Jupyter Notebook**: For running ML experiments interactively.
* **Google Colab**: Free platform with GPU support for ML.
* **GitHub**: Version control for your projects.

**7. Learn SQL (Optional but Highly Recommended)**

* Data retrieval is essential for ML. Learn basic queries, joins, and aggregations.

**8. Cloud Platforms (Optional for Deployment)**

Once you’re comfortable with ML basics, explore cloud services like:

* AWS (SageMaker for ML).
* Google Cloud (Vertex AI).
* Microsoft Azure.

**Suggested Roadmap to Get Started**

1. **Start with Python and basic math.**
2. **Learn data preprocessing and visualization (NumPy, Pandas, Matplotlib).**
3. **Dive into statistics and probability.**
4. **Learn ML basics (linear regression, logistic regression, decision trees).**
5. **Practice on datasets (e.g., Kaggle, UCI Machine Learning Repository).**
6. **Explore advanced topics (neural networks, deep learning).**

**Resources to Learn From**

1. **Books**:
   * *"Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow"* by Aurélien Géron.
   * *"Deep Learning"* by Ian Goodfellow (for advanced learners).
2. **Online Courses**:
   * [Andrew Ng’s Machine Learning Course (Coursera)](https://www.coursera.org/learn/machine-learning).
   * [Deep Learning Specialization (Coursera)](https://www.coursera.org/specializations/deep-learning).
   * Google ML Crash Course.
3. **Practice Platforms**:
   * Kaggle.
   * Google Colab (free GPU for training models).

**What to Do Next?**

Start small! Practice simple projects like:

* Predicting house prices (regression).
* Classifying emails as spam or not (classification).
* Clustering customers into groups (unsupervised learning).

Let me know if you'd like help with specific topics or project ideas! 😊