

You have **2 free member-only stories left** this month. [Sign up](#) for Medium and get an extra one.

★ Member-only story

FUNDAMENTALS

Understanding the Covariance Matrix

All you need to know about variance, standard deviation, correlation, and covariance



Marvin Lanhенке · Follow

Published in Towards Data Science

6 min read · Dec 29, 2021

Listen

Share

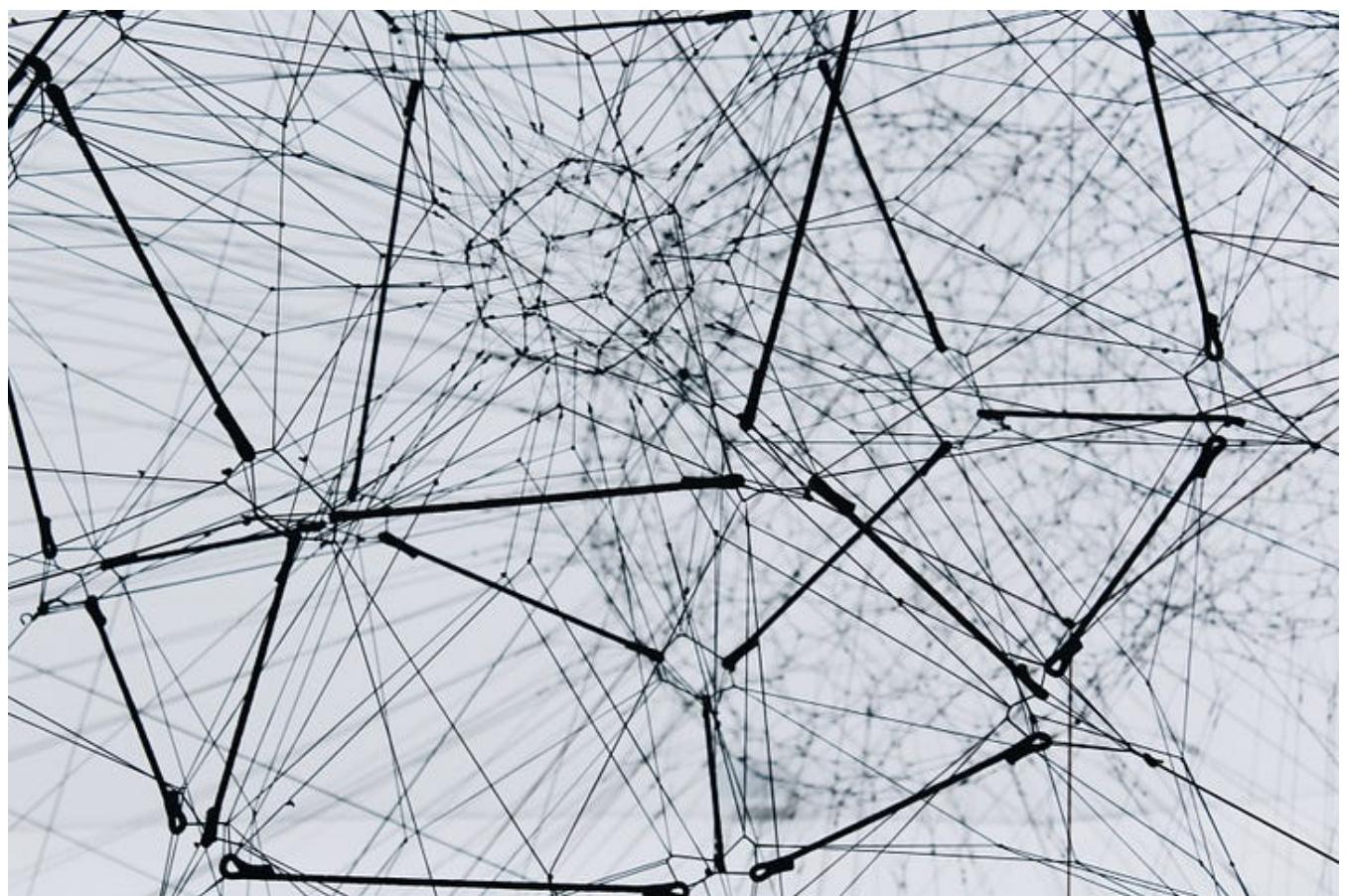


Photo by [Alina Grubnyak](#) on [Unsplash](#)

Linear algebra is one of the fundamentals of machine learning and is considered the '*mathematics of data*'. While I personally enjoy many aspects of linear algebra, some concepts are not easy to grasp at first. I often struggled to imagine the real-world application or the actual benefit of some concepts.

The covariance matrix, however, tells a completely different story.

The concepts of covariance and correlation bring some aspects of linear algebra to life. Algorithms, like PCA for example, depend heavily on the computation of the covariance matrix, which plays a vital role in obtaining the principal components.

Implementing PCA From Scratch

Brushing up your linear algebra skills with just Python and NumPy

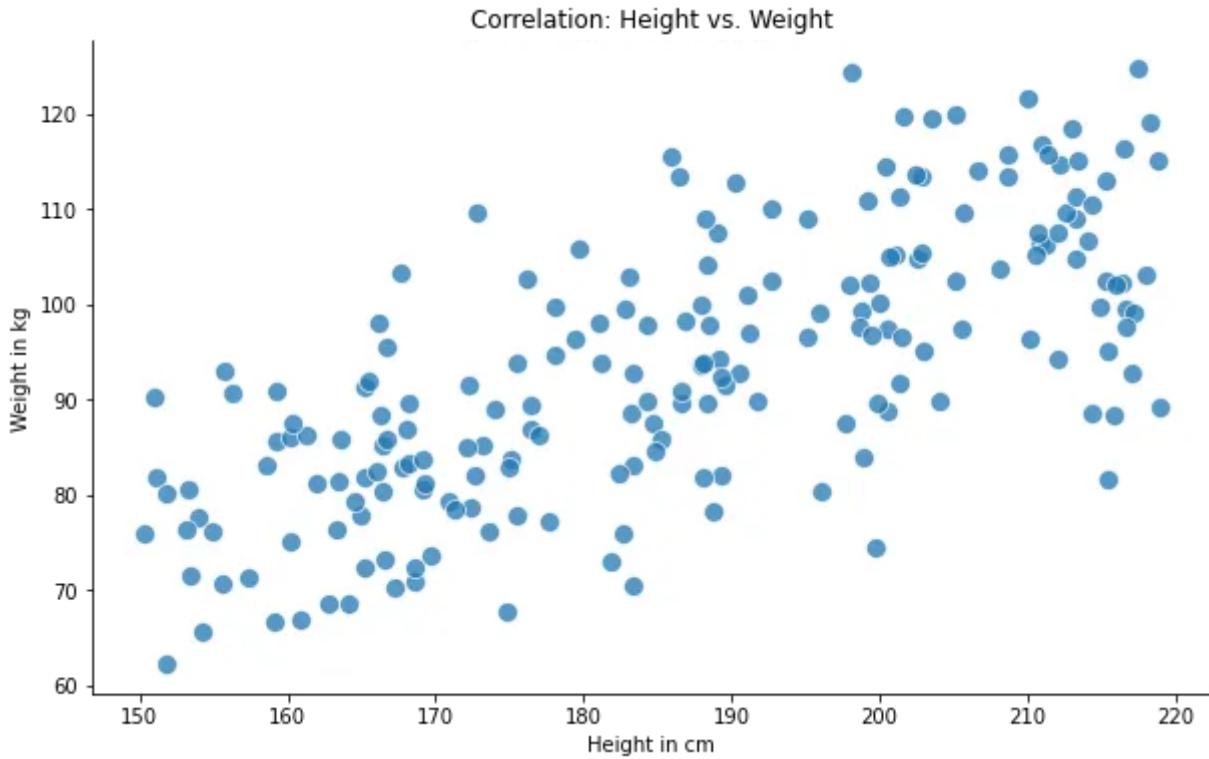
[towardsdatascience.com](https://towardsdatascience.com/implementing-pca-from-scratch-5a2a2f3e0a2c)

In the following sections, we are going to learn about the covariance matrix, how to calculate and interpret it. But first of all, we need to learn about the related concepts, the basics, allowing us to gain a deeper understanding.

What is correlation and what does it tell us?

Correlation analysis aims to identify commonalities between variables.

Let's imagine, we measure the variables '*height*' and '*weight*' from a random group of people. In general, we would expect the taller people to weigh more than the shorter people. A scatterplot of such a relation could look like this:



A scatterplot of the variables height and weight [Image by Author]

By looking at the plot above, we can clearly tell that both variables are related. Correlation, or more specifically the correlation coefficient, provides us with a statistical measure to quantify that relation.

The coefficient ranges from minus one to positive one and can be interpreted as the following:

1. Positive correlation — tells us that both variables move in the same direction e.g. both values are increasing simultaneously.
2. Negative correlation — describes inversely correlated variables. Meaning, if one variable is increasing the other is decreasing or vice-versa.
3. A correlation coefficient of zero shows that there is no relationship at all.

Note: The correlation coefficient is limited to linearity and therefore won't quantify any non-linear relations. To measure non-linear relationships one can use other approaches such as mutual information or transforming the variable.

So why do we even care about correlation?

It turns out that the correlation coefficient and the covariance are basically the same concepts and are therefore closely related. The correlation coefficient is

simply the normalized version of the covariance bound to the range [-1,1].

Both concepts rely on the same foundation: the variance and the standard deviation.

Introducing variance and standard deviation

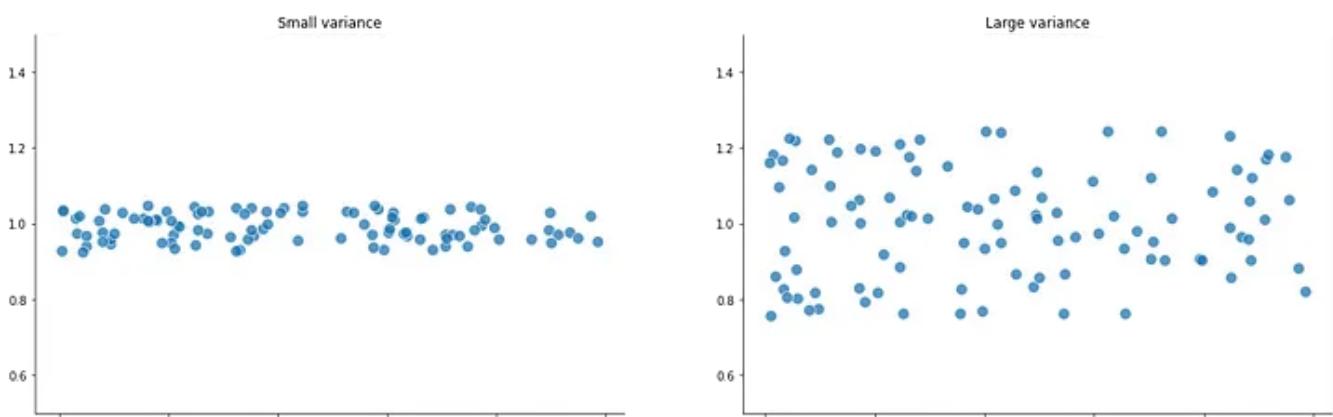
Variance as a measure of dispersion, tells us how different or how spread out our data values are.

We can compute the variance by taking the average of the squared difference between each data value and the mean, which is, loosely speaking, just the distance of each data point to the center.

$$\sigma^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})$$

By looking at the equation, we can already tell, that when all data values are close to the mean the variance will be small. If the data points are far away from the center, the variance will be large.

Let's take a look at two examples to make things a bit more tangible.



An example of small vs. large variance data [Image by Author]

Once we know the variance, we also know the standard deviation. It is simply the square root of the variance.

$$\sigma = \sqrt{\sigma^2}$$

Now that we know the underlying concepts, we can tie things together in the next section.

Covariance and the covariance matrix

Assume, we have a dataset with two features and we want to describe the different relations within the data. The concept of covariance provides us with the tools to do so, allowing us to measure the variance between two variables.

We can calculate the covariance by slightly modifying the equation from before, basically computing the variance of two variables with each other.

$$C_{x,y} = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})$$

If we mean-center our data before, we can simplify the equation to the following:

$$C_{x,y} = \frac{x^T y}{n-1}$$

Once simplified, we can see that the calculation of the covariance is actually quite simple. It is just the dot product of two vectors containing data.

Now imagine, a dataset with three features x, y, and z. Computing the covariance matrix will yield us a 3 by 3 matrix. This matrix contains the covariance of each feature with all the other features and itself. We can visualize the covariance matrix like this:

$$C(x, y, z) = \begin{bmatrix} var_x & covar_{x,y} & covar_{x,z} \\ covar_{y,x} & var_y & covar_{y,z} \\ covar_{z,x} & covar_{z,y} & var_z \end{bmatrix}$$

Example based on [Implementing PCA From Scratch](#)

The covariance matrix is symmetric and feature-by-feature shaped. The diagonal contains the variance of a single feature, whereas the non-diagonal entries contain the covariance.

We already know how to compute the covariance matrix, we simply need to exchange the vectors from the equation above with the mean-centered data matrix.

$$C_{x,y} = \frac{X^T X}{n-1}$$

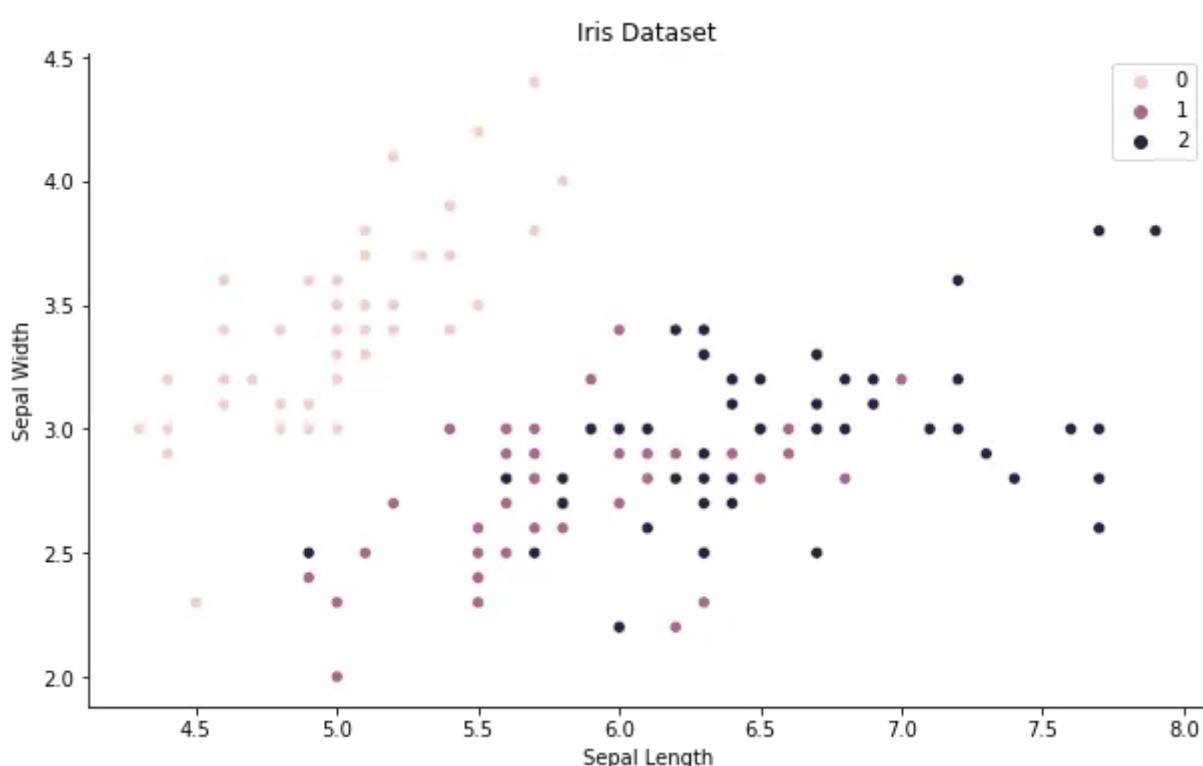
Once calculated, we can interpret the covariance matrix in the same way as described earlier, when we learned about the correlation coefficient.

Applying our knowledge

Now that we've finished the groundwork, let's apply our knowledge.

For testing purposes, we will use the iris dataset. The dataset consists of 150 samples with 4 different features (*Sepal Length*, *Sepal Width*, *Petal Length*, *Petal Width*)

Let's take a first glance at the data by plotting the first two features in a scatterplot.



An overview of the iris dataset by plotting the first two features [Image by Author]

Our goal is to ‘manually’ compute the covariance matrix. Hence, we need to mean-center our data before. In order to do that, we define and apply the following function:

```

1 def standardize_data(X):
2     numerator = X - np.mean(X, axis=0)
3     denominator = np.std(X, axis=0)
4     return numerator / denominator
5
6 X_scl = standardize_data(X)
7
8 print('Mean:', np.mean(X_scl))
9 print('Std:', np.std(X_scl))

```

covar_mean_center.py hosted with ❤ by GitHub

[view raw](#)

Note: We standardize the data by subtracting the mean and dividing it by the standard deviation.

Running the code above, standardizes our data and we obtain a mean of zero and a standard deviation of one as expected.

Next, we can compute the covariance matrix.

```

1 def get_covariance_matrix(X, ddof=0):
2     n_samples = X.shape[0]
3     C = np.dot(X.T, X) / (n_samples - ddof)
4     return C
5
6 C = get_covariance_matrix(X_scl, ddof=0)
7
8 print(C.shape)

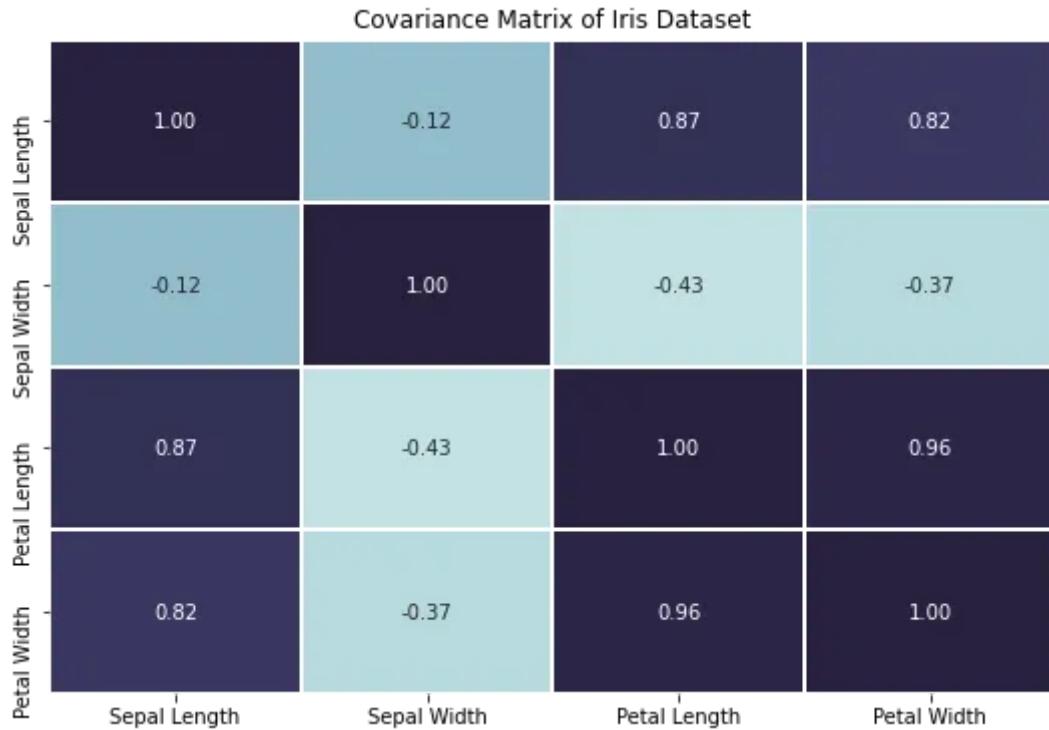
```

covar_compute.py hosted with ❤ by GitHub

[view raw](#)

Note: The same computation can be achieved with NumPy’s built-in function `numpy.cov(x)`.

Our covariance matrix is a 4 by 4 matrix, shaped feature-by-feature. We can visualize the matrix and the covariance by plotting it like the following:



Covariance matrix plotted as a heatmap [Image by Author]

We can clearly see a lot of correlation among the different features, by obtaining high covariance or correlation coefficients. For example, the petal length seems to be highly positively correlated with the petal width, which makes sense intuitively – if the petal is longer it is probably also wider.

Conclusion

In this article, we learned how to compute and interpret the covariance matrix. We also covered some related concepts such as variance, standard deviation, covariance, and correlation.

The covariance matrix plays a central role in the [principal component analysis](#). Implementing or computing it in a more manual approach ties a lot of important pieces together and breathes life into some linear algebra concepts.

Thank you for reading! Make sure to stay connected & follow me here on [Medium](#), [Kaggle](#), or just say 'Hi' on [LinkedIn](#)

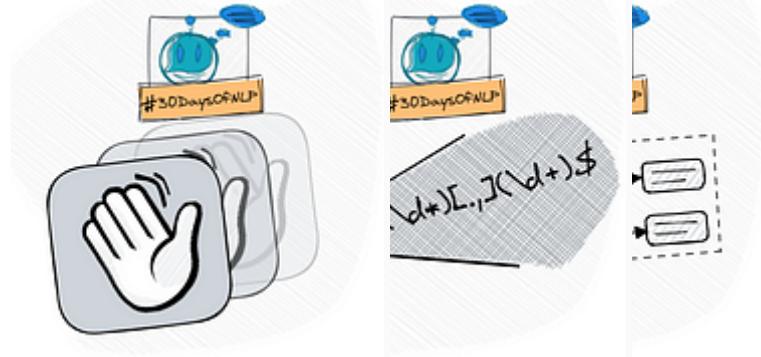


Marvin Lanhenke

#30DaysOfNLP

[View list](#)

30 stories



Enjoyed the article? Become a [Medium member](#) and continue learning with no limits. I'll receive a portion of your membership fee if you use the following link, at no extra cost to you.

[Open in app](#)[Sign up](#)[Sign In](#)

As a Medium member, a portion of your membership fee goes to writers you read, and you get full access to every story...

medium.com

References / Further Material:

- [Implementing PCA From Scratch](#)
- Mike X Cohen, PhD. Linear Algebra: Theory, Intuition, Code.

[Machine Learning](#)[Data Science](#)[Statistics](#)[Fundamentals](#)[Covariance Matrix](#)

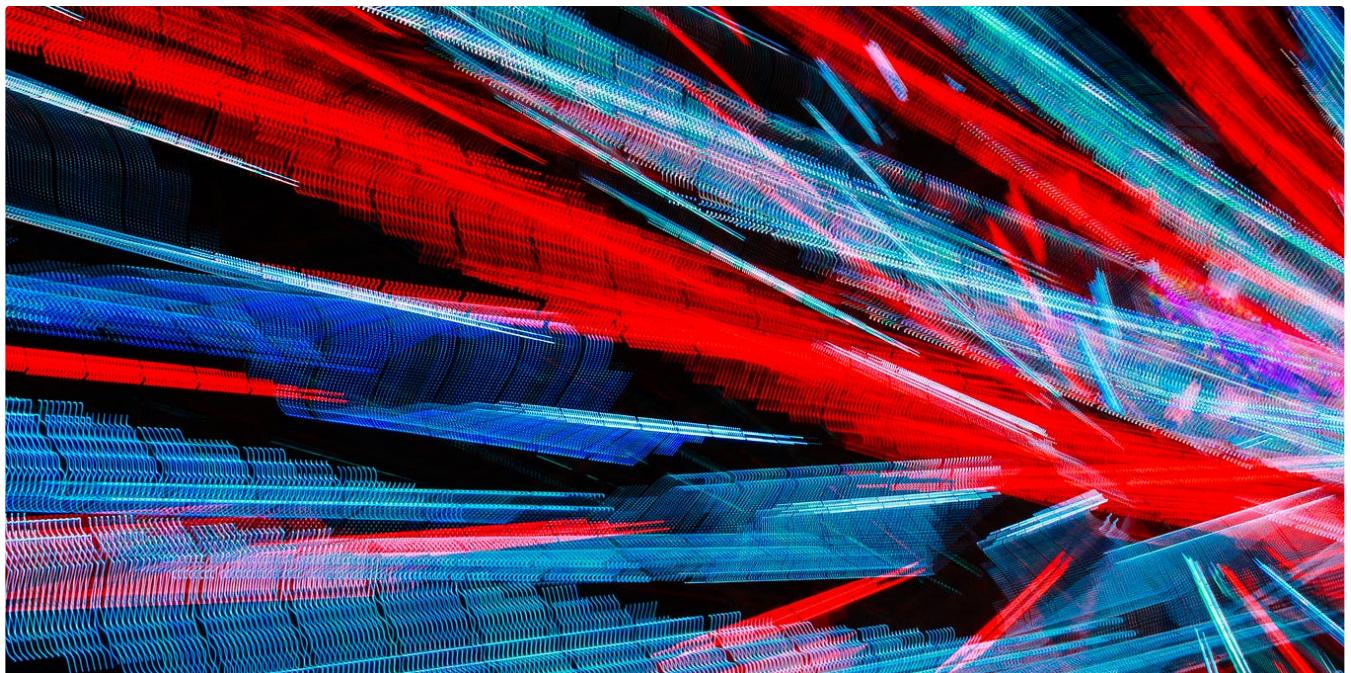
[Follow](#)

Written by Marvin Lanhенке

772 Followers · Writer for Towards Data Science

Business Analyst. Solutions Architect. Self-Taught. Hands-On. Writing about Software Architecture & Engineering. Say Hi @ linkedin.com/in/marvinlanhenke/

More from Marvin Lanhенке and Towards Data Science



Marvin Lanhенке in Towards Data Science

Setting Up Apache Airflow with Docker-Compose in 5 Minutes

Create a development environment and start building DAGs

★ · 6 min read · May 31, 2022

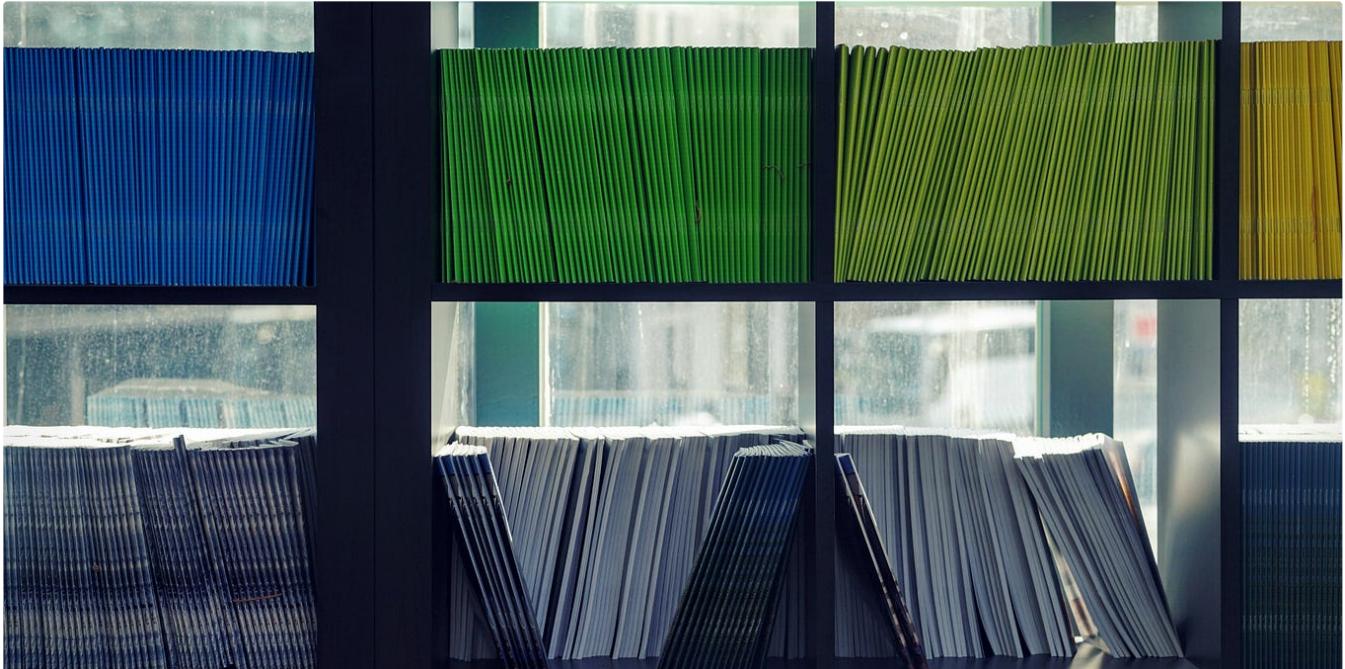


235



1



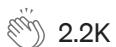


 Jacob Marks, Ph.D. in Towards Data Science

How I Turned My Company's Docs into a Searchable Database with OpenAI

And how you can do the same with your docs

15 min read · Apr 25



2.2K



27



 Leonie Monigatti in Towards Data Science

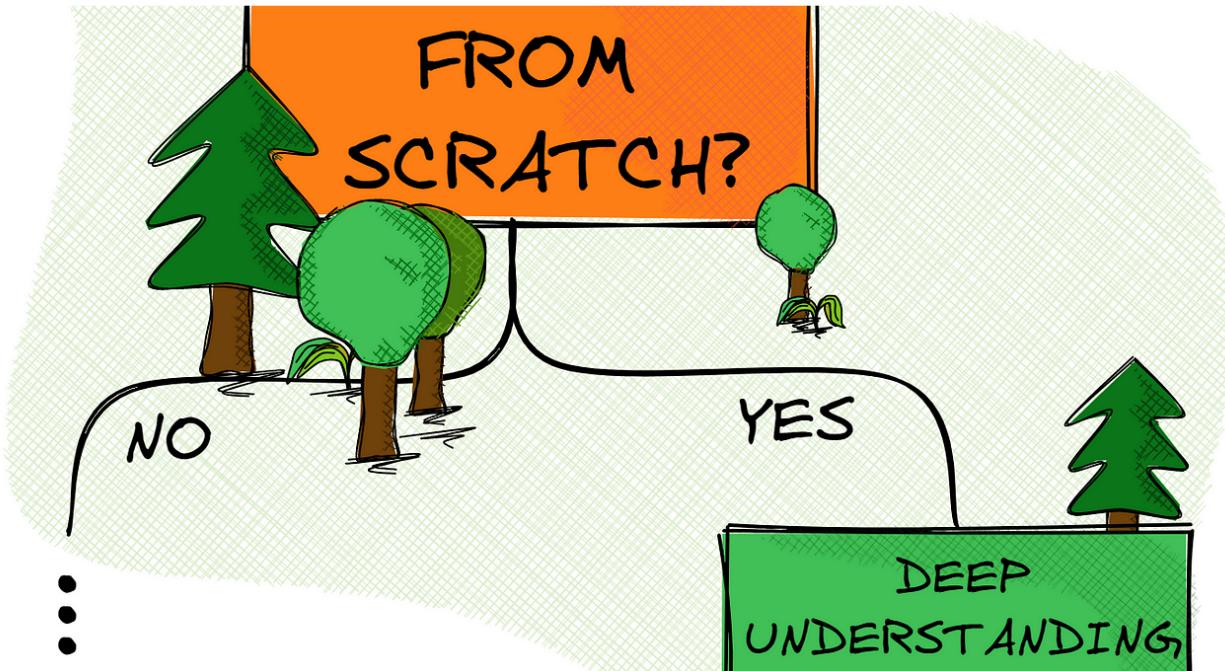
Getting Started with LangChain: A Beginner's Guide to Building LLM-Powered Applications

A LangChain tutorial to build anything with large language models in Python

★ · 12 min read · Apr 25

👏 1.1K

💬 12



Marvin Lanhенке in Towards Data Science

Implementing a Decision Tree From Scratch

Recursively refine your understanding by implementing a decision tree with just Python and NumPy

★ · 8 min read · Feb 2, 2022

👏 69

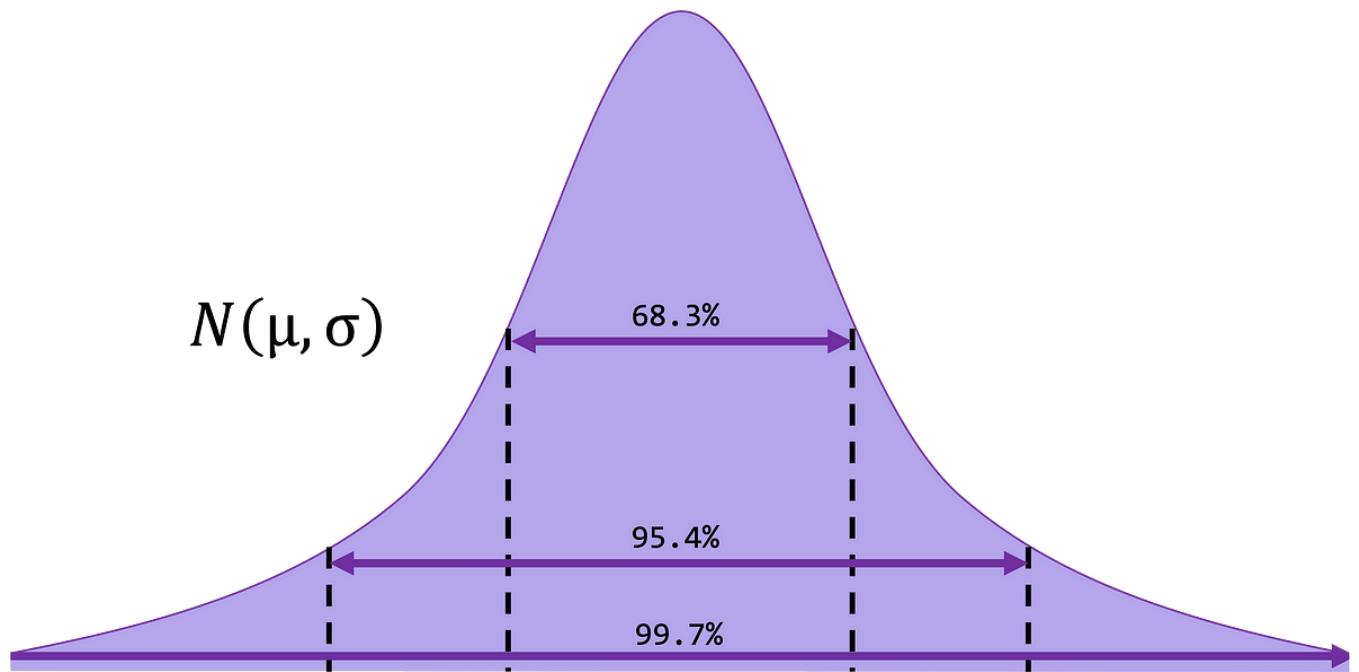
💬 6



See all from Marvin Lanhенке

See all from Towards Data Science

Recommended from Medium



 Adrienne Kline in Towards Data Science

Statistics Bootcamp 7: Balancing Type I and II Errors

Learn the math and methods behind the libraries you use daily as a data scientist

★ · 12 min read · Nov 23, 2022

 124

 1





Chao De-Yu in Towards Data Science

Two-Way ANOVA Test, with Python

The Complete Beginner's Guide to perform Two-Way ANOVA Test (with code!)

◆ · 6 min read · Jan 5

131

1



Lists



What is ChatGPT?

9 stories · 23 saves



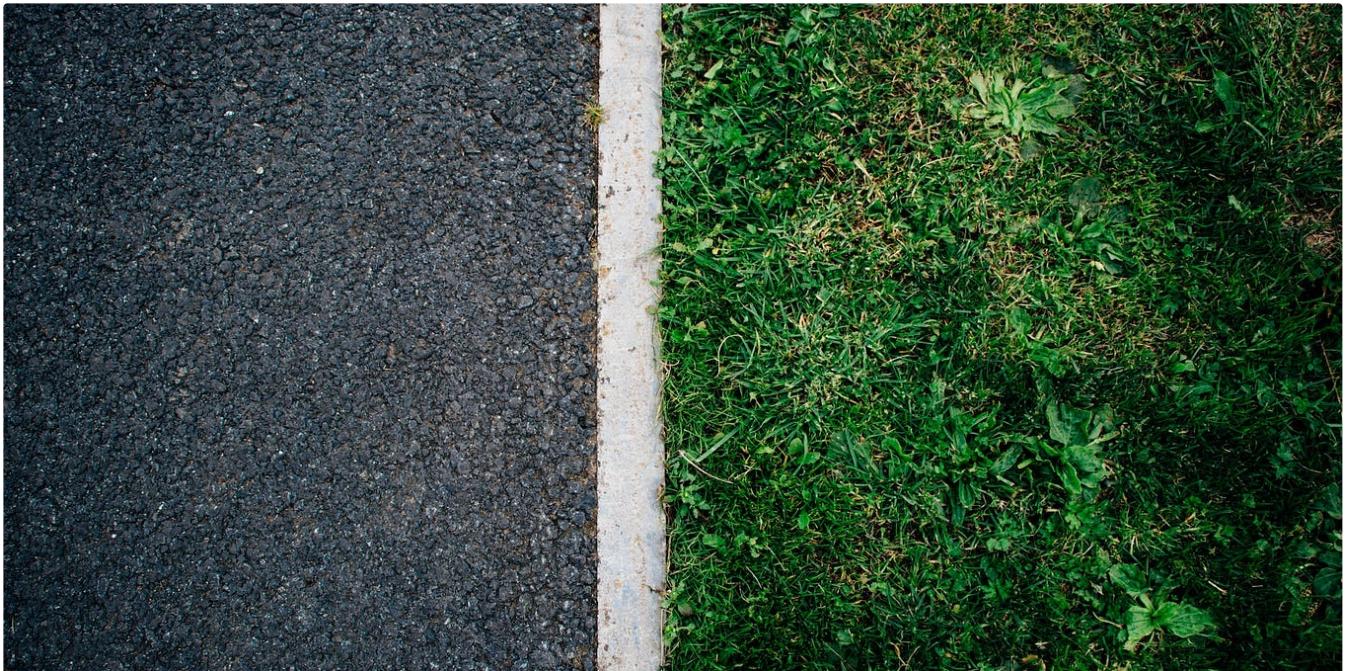
Stories to Help You Level-Up at Work

19 stories · 17 saves



Staff Picks

300 stories · 62 saves



Rukshan Pramoditha in Towards Data Science

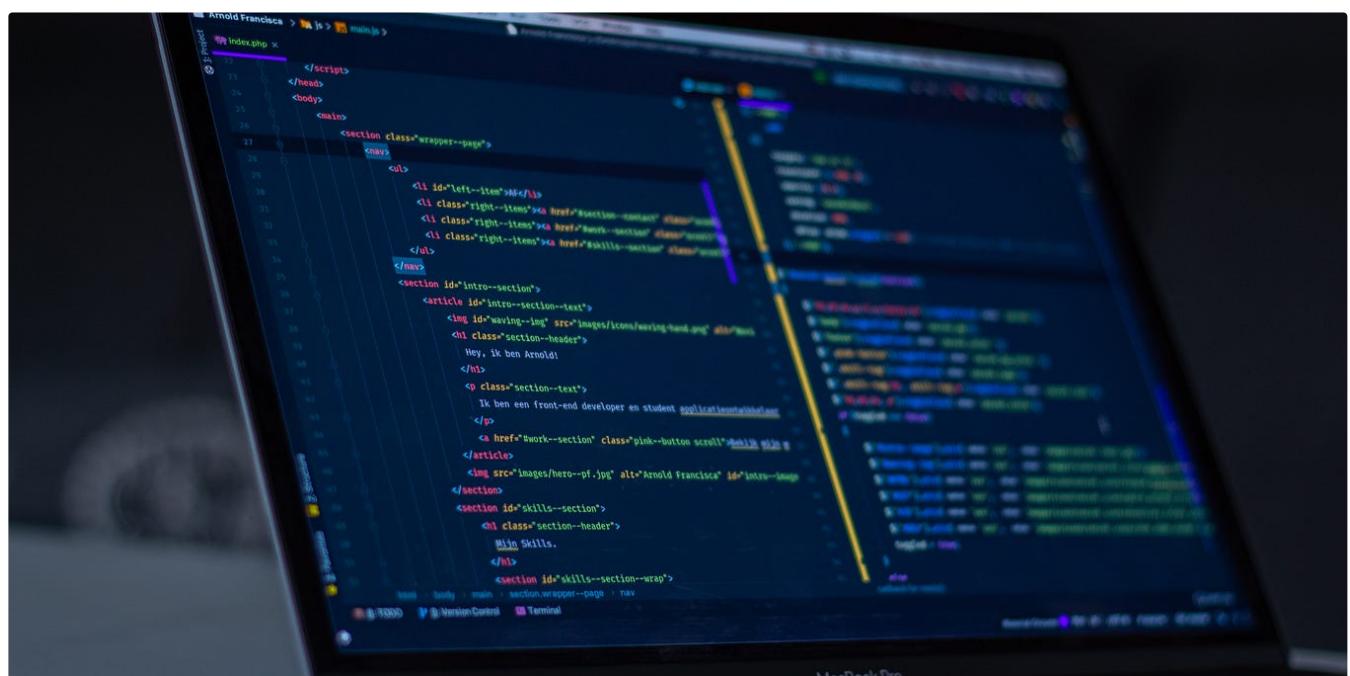
LDA Is More Effective than PCA for Dimensionality Reduction in Classification Datasets

Linear discriminant analysis (LDA) for dimensionality reduction while maximizing class separability

◆ · 11 min read · Dec 29, 2022



92



Carla Martins

Statistics for Data science: Comparing The Distribution of Two Categorical Variables

Chi-Square test

★ · 4 min read · Dec 6, 2022

👏 65



Dr. Mandar Karhade, MD, PhD. in Geek Culture

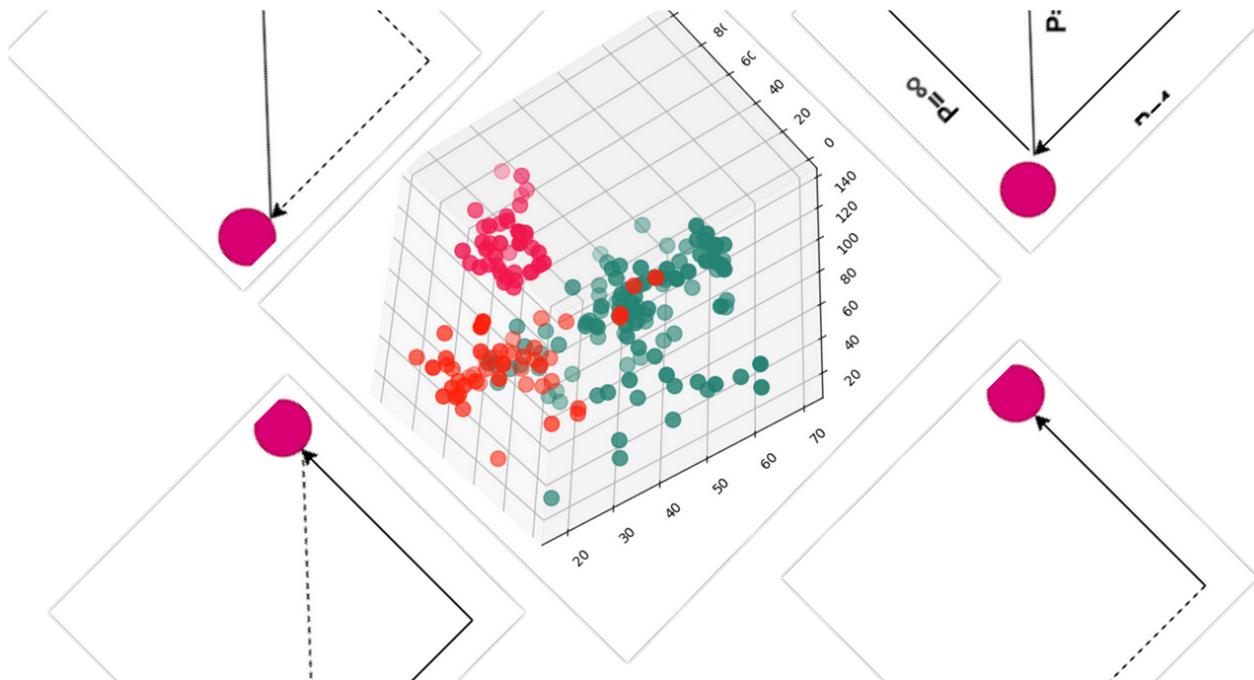
Everything about Linear Discriminant Analysis (LDA)

All you need to know to conduct Linear Discriminant Analysis.

★ · 6 min read · Dec 10, 2022

👏 23





 Md. Zubair in Towards Data Science

K-means Clustering from Scratch

K-means: The Best ML Algorithm to Cluster Data

★ · 9 min read · Dec 19, 2022

 89 



See more recommendations