Big Data Analytics Project:

EL NINO DATA ANALYSIS

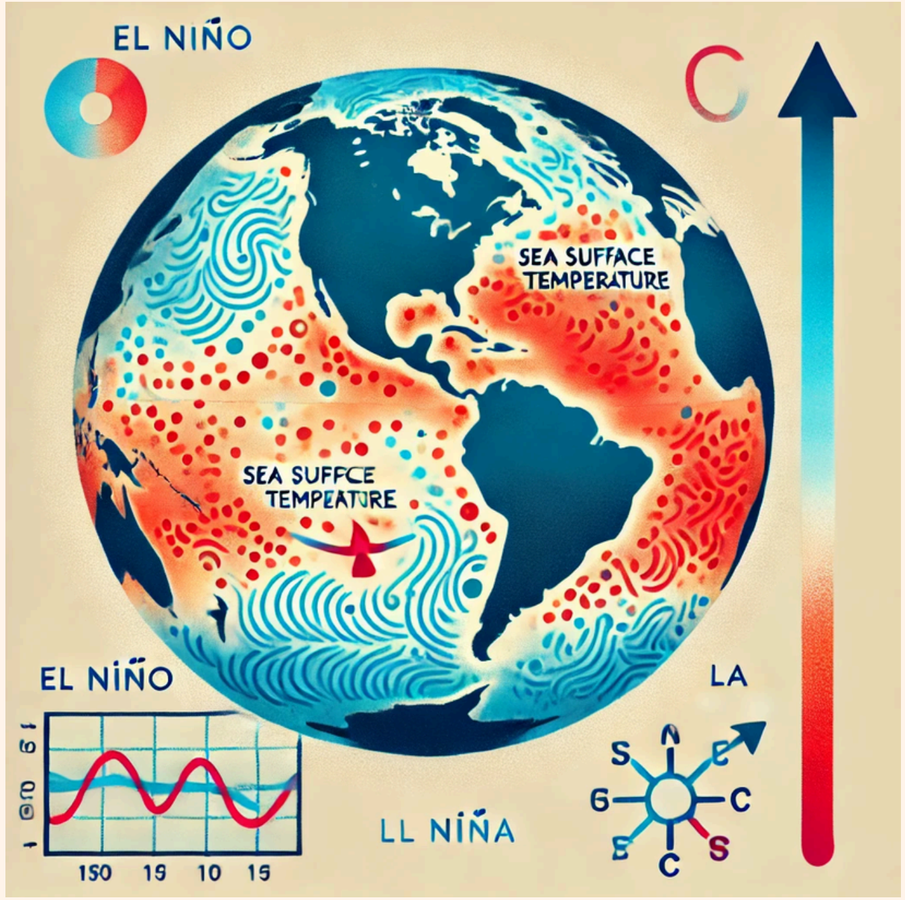
5th Feb, 2025

Presented by

Annisaa Nurfirdausi

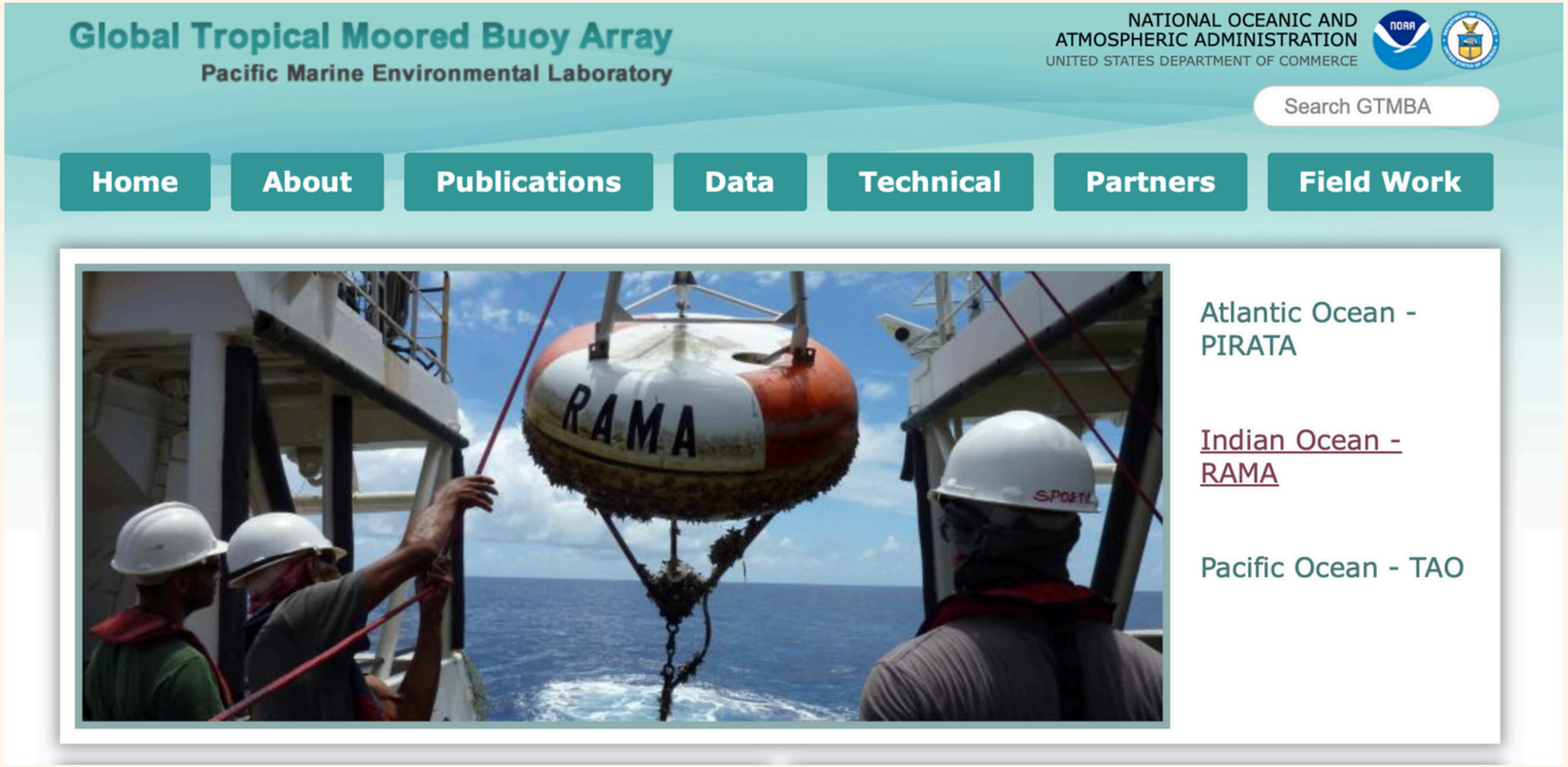
INTRODUCTION AND MOTIVATION

- El Niño is a complex climate phenomenon characterized by the periodic warming of sea surface temperatures (SST) [1].
- We aim to analyse SST based on the available atmospheric and oceanographic variables using data processing and machine learning techniques.
- We will use PySpark to handle large-scale data processing,



[1] https://www.pmel.noaa.gov/

DATASET SOURCE



The TAO array consists of around 70 moored buoys across the equatorial Pacific, measuring oceanographic and meteorological variables vital for detecting and predicting climate variations[1]

[1] https://www.pmel.noaa.gov/

DATASET: VARIABLE CHARACTERISTICS

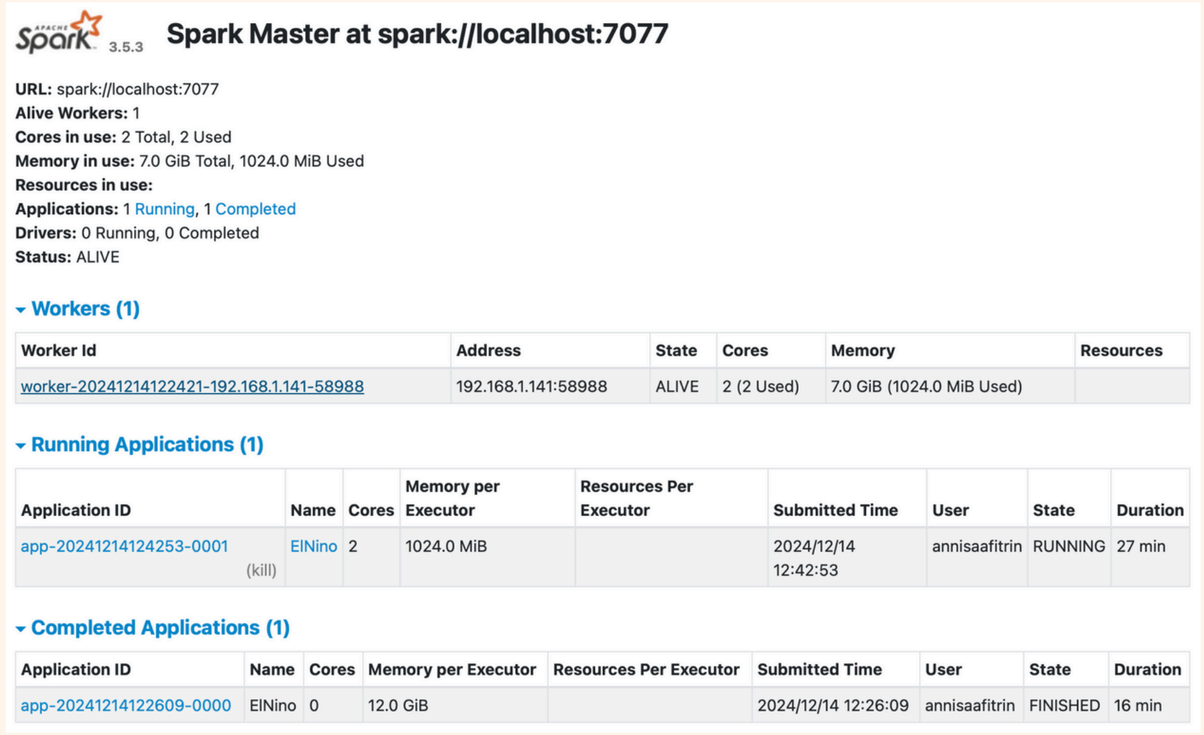
- **Date:** Time of the reading
- **Latitude & Longitude**: Buoy locations (latitude within 1°, longitude varies up to 5°)
- **Zonal Winds**: -10 m/s to 10 m/s
- **Meridional Winds**: -10 m/s to 10 m/s
- **Relative Humidity**: 70% to 90%
- **Air Temperature:** 20°C to 30°C
- **Sea Surface Temperature**: 20°C to 30°C (positive linear relationship with air temp)

Additional characteristics:

- **Time period:** March 7th 1980 - June 5th 1998
- **Total records:** 178k
- There are missing values

observation	date	latitude	longitude	zonal_winds	meridional_winds	humidity	air_temperature	sea_surface_temperature	
0	0	0	0	25163	25162	65761	18237	17007	

METHODOLOGY:
ARCHITECTURE



Spark 3.5.3

Spark Master at spark://localhost:7077

URL: spark://localhost:7077
Alive Workers: 1
Cores in use: 2 Total, 2 Used
Memory in use: 7.0 GiB Total, 1024.0 MiB Used
Resources in use:
Applications: 1 Running, 1 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers (1)

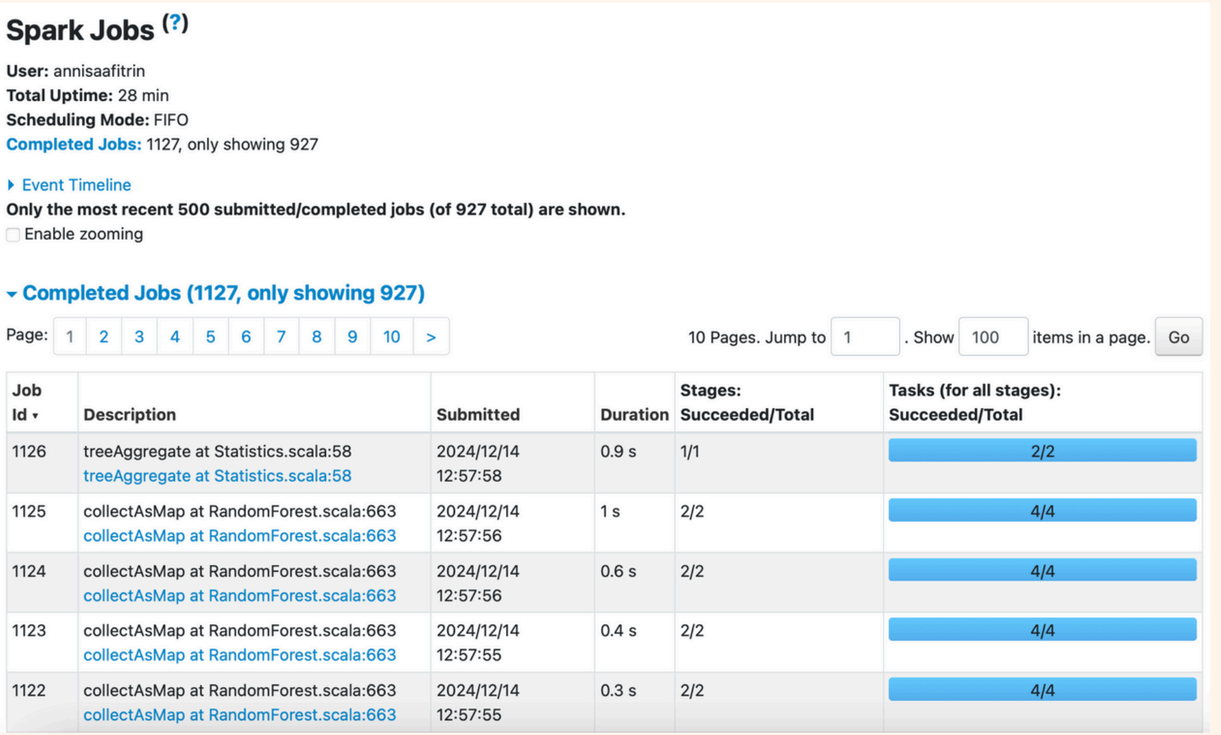
Worker Id	Address	State	Cores	Memory	Resources
worker-20241214122421-192.168.1.141-58988	192.168.1.141:58988	ALIVE	2 (2 Used)	7.0 GiB (1024.0 MiB Used)	

Running Applications (1)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration
app-20241214124253-0001 (kill)	ElNino	2	1024.0 MiB		2024/12/14 12:42:53	annisaafitrin	RUNNING	27 min

Completed Applications (1)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration
app-20241214122609-0000	ElNino	0	12.0 GiB		2024/12/14 12:26:09	annisaafitrin	FINISHED	16 min



Spark Jobs (?)

User: annisaafitrin
Total Uptime: 28 min
Scheduling Mode: FIFO
Completed Jobs: 1127, only showing 927

Event Timeline
Only the most recent 500 submitted/completed jobs (of 927 total) are shown.
Enable zooming

Completed Jobs (1127, only showing 927)

Page: 1 2 3 4 5 6 7 8 9 10 > 10 Pages. Jump to 1 . Show 100 items in a page. Go

Job Id	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
1126	treeAggregate at Statistics.scala:58 treeAggregate at Statistics.scala:58	2024/12/14 12:57:58	0.9 s	1/1	2/2
1125	collectAsMap at RandomForest.scala:663 collectAsMap at RandomForest.scala:663	2024/12/14 12:57:56	1 s	2/2	4/4
1124	collectAsMap at RandomForest.scala:663 collectAsMap at RandomForest.scala:663	2024/12/14 12:57:56	0.6 s	2/2	4/4
1123	collectAsMap at RandomForest.scala:663 collectAsMap at RandomForest.scala:663	2024/12/14 12:57:55	0.4 s	2/2	4/4
1122	collectAsMap at RandomForest.scala:663 collectAsMap at RandomForest.scala:663	2024/12/14 12:57:55	0.3 s	2/2	4/4

Master URL: spark://localhost:7077

Number of Executors: 2

Total Cores: 2

METHODOLOGY:

FEATURE ENGINEERING

obs	year	month	day	date	latitude	longitude	zon.winds	mer.winds	humidity	air temp.	s.s.temp.
1	80	3	7	800307	-0.02	-109.46	-6.8	0.7	.	26.14	26.24
2	80	3	8	800308	-0.02	-109.46	-4.9	1.1	.	25.66	25.97
3	80	3	9	800309	-0.02	-109.46	-4.5	2.2	.	25.69	25.28
4	80	3	10	800310	-0.02	-109.46	-3.8	1.9	.	25.57	24.31
5	80	3	11	800311	-0.02	-109.46	-4.2	1.5	.	25.3	23.19

adding Ocean column:

"West Pacific": longitude < -170 or longitude > 160

"East Pacific": -120 < longitude < -70

adding Hemisphere column:

South: latitude < 0

North: latitude > 0

adding ElNino period:

The first El Niño period is from May 1, 1982, to June 30, 1983 [1]

The second El Niño period is from May 1, 1997, to June 30, 1998 [2]

observation	date	latitude	longitude	zonal_winds	meridional_winds	humidity	air_temperature	sea_surface_temperature	El Nino	Ocean	Hemisphere
1	1980-03-07	-0.02	-109.46	-6.8	0.7	.	26.14	26.24	No	East Pacific	South
2	1980-03-08	-0.02	-109.46	-4.9	1.1	.	25.66	25.97	No	East Pacific	South
3	1980-03-09	-0.02	-109.46	-4.5	2.2	.	25.69	25.28	No	East Pacific	South
4	1980-03-10	-0.02	-109.46	-3.8	1.9	.	25.57	24.31	No	East Pacific	South
5	1980-03-11	-0.02	-109.46	-4.2	1.5	.	25.3	23.19	No	East Pacific	South

[1] [https://www.nature.com/articles/305016a0.pdf]

[2] https://en.wikipedia.org/wiki/1997–98_El_Niño_event

METHODOLOGY:

DATA CLEANING - MISSING VALUE HANDLING

observation	date	latitude	longitude	zonal_winds	meridional_winds	humidity	air_temperature	sea_surface_temperature
0	0	0	0	25163	25162	65761	18237	17007

These values are cyclic/periodic

Filling the missing values with Imputer - most frequent

These values are continues

Filling the missing values with mean

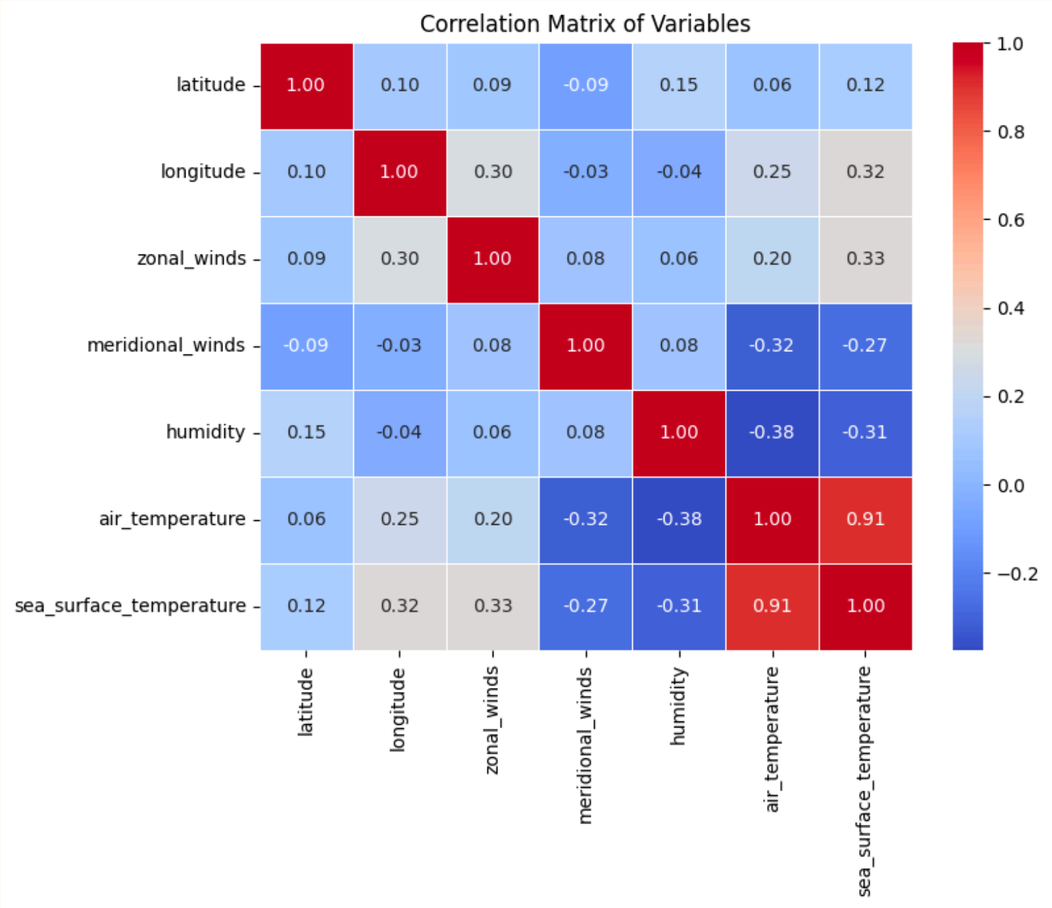
dropping rows with missing values

There are 112319 rows in the dataset after removing rows

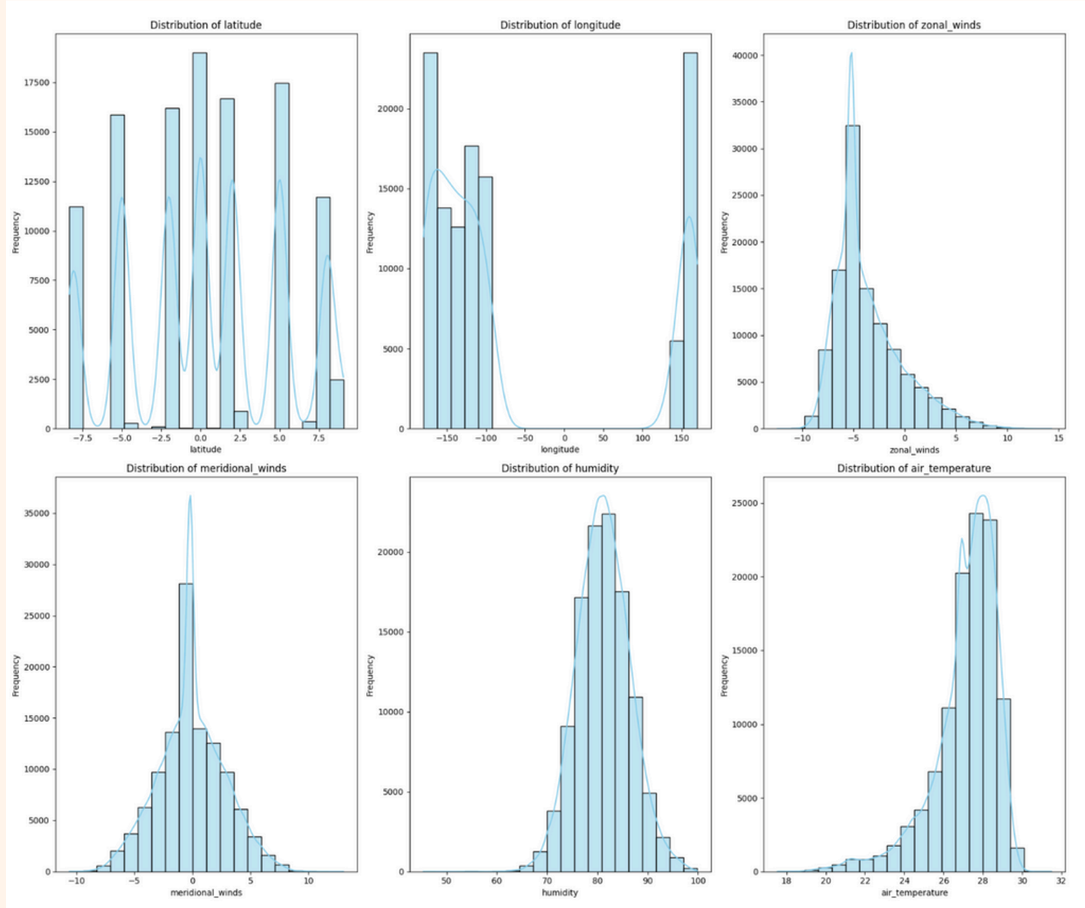
METHODOLOGY:

EXPLORATORY DATA ANALYSIS (EDA)

Correlation Matrix



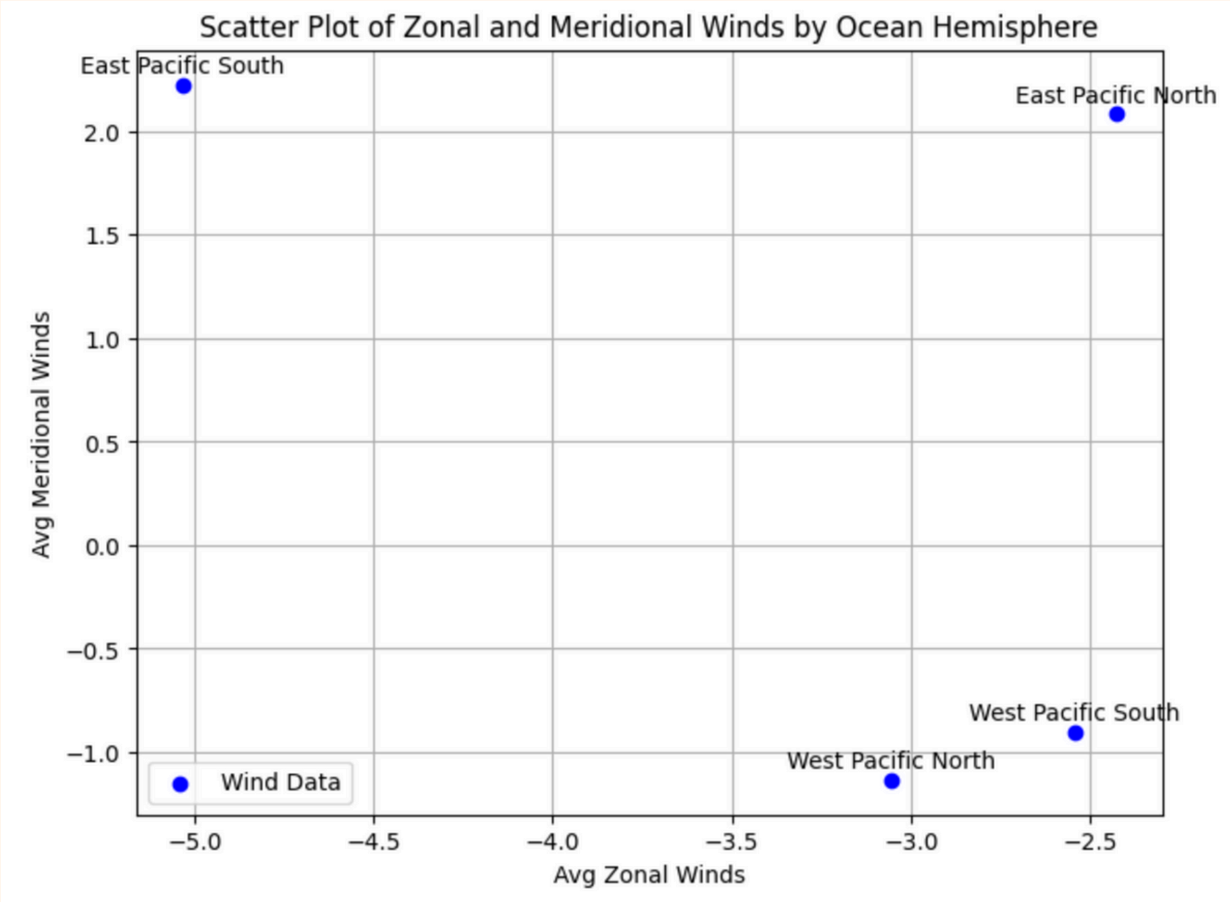
Data Distribution



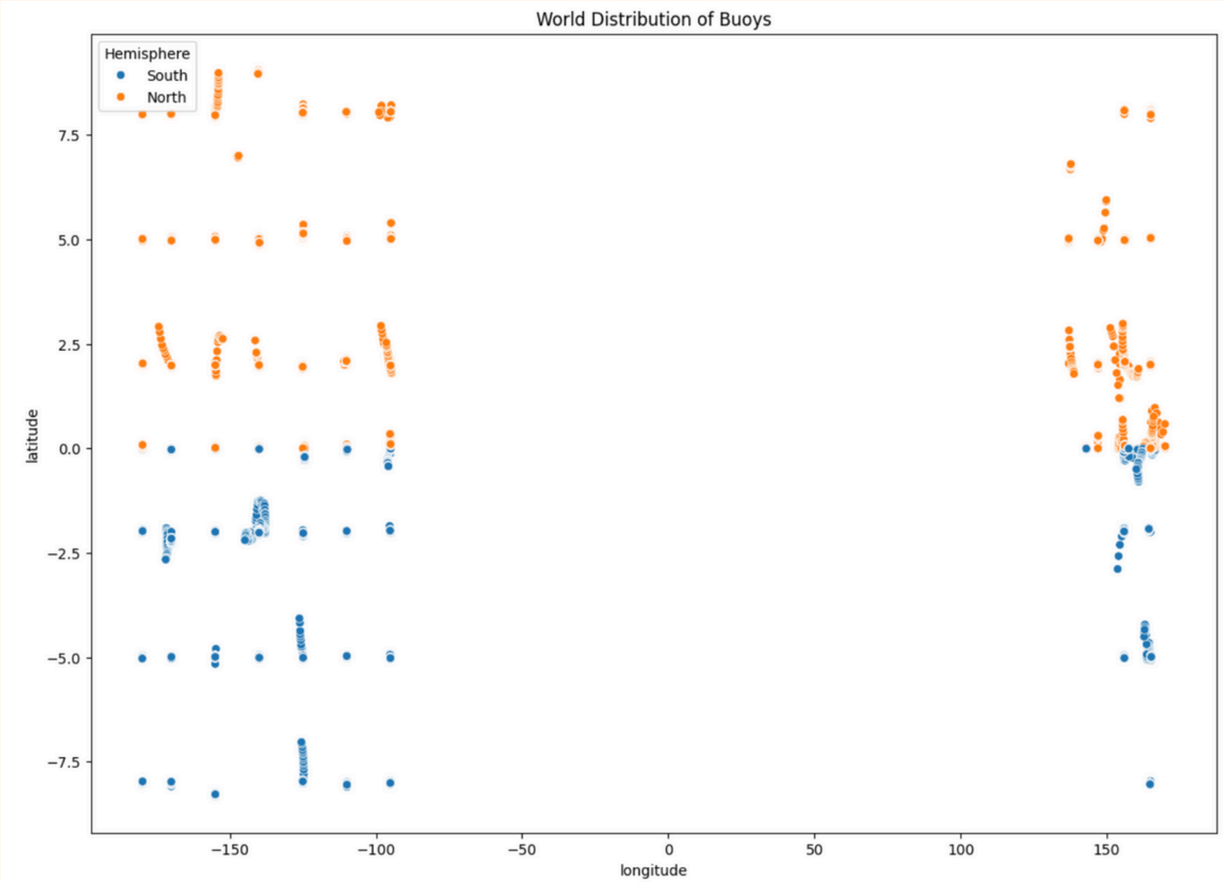
METHODOLOGY:

EXPLORATORY DATA ANALYSIS (EDA)

Wind



Buoys Distribution



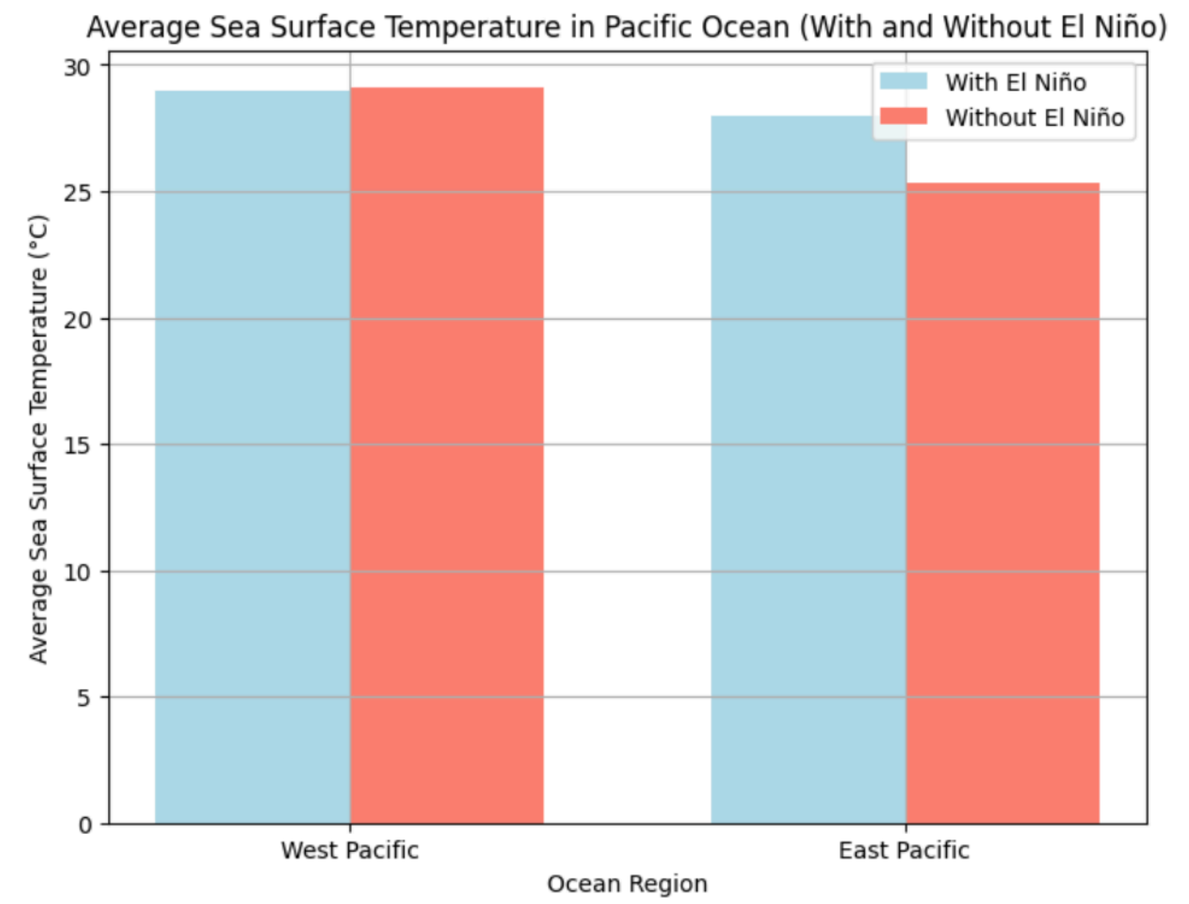
Zonal: West - East winds

Meridional: North - South winds

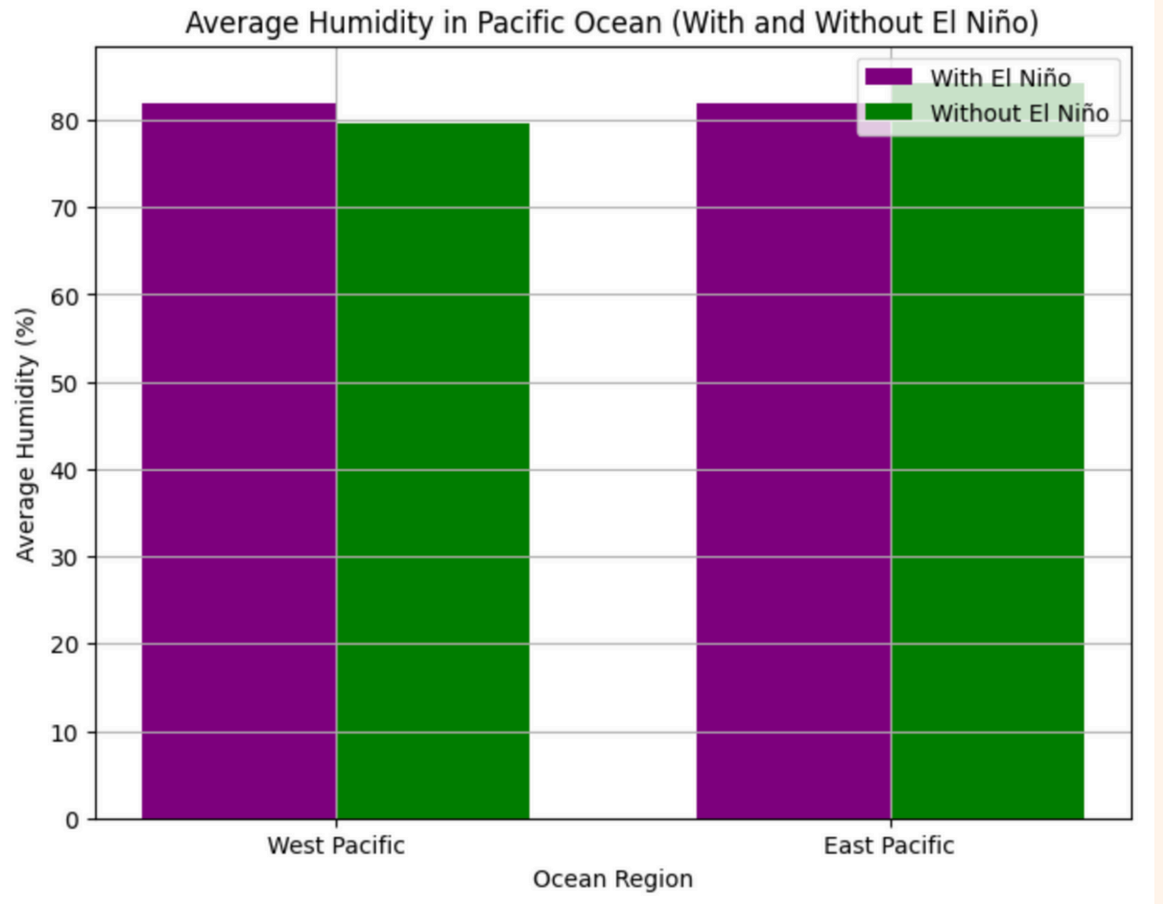
METHODOLOGY:

EXPLORATORY DATA ANALYSIS (EDA)

Sea Sufrace Temperature



Humidity



METHODOLOGY:

MACHINE LEARNING ALGORITHMS

1. Linear Regression

Linear regression predicts a target variable by fitting a straight line to the data

```
lr_paramGrid = ParamGridBuilder() \
    .addGrid(lr.regParam, [0.01, 0.1, 1.0]) \
    .addGrid(lr.elasticNetParam, [0.0, 0.5, 1.0]) \
    .build()

lr_crossval = CrossValidator(estimator=lr,
                             estimatorParamMaps=lr_paramGrid,
                             evaluator=evaluator,
                             numFolds=3)
```

2. Decision Tree Regression

Random forest regression uses an ensemble of decision trees to predict a target variable, averaging their outputs.

```
dt_paramGrid = ParamGridBuilder() \
    .addGrid(dt.maxDepth, [5, 10]) \
    .addGrid(dt.maxBins, [16, 32, 64]) \
    .build()

dt_crossval = CrossValidator(estimator=dt,
                             estimatorParamMaps=dt_paramGrid,
                             evaluator=evaluator,
                             numFolds=3)
```

3. Random Forest Regression

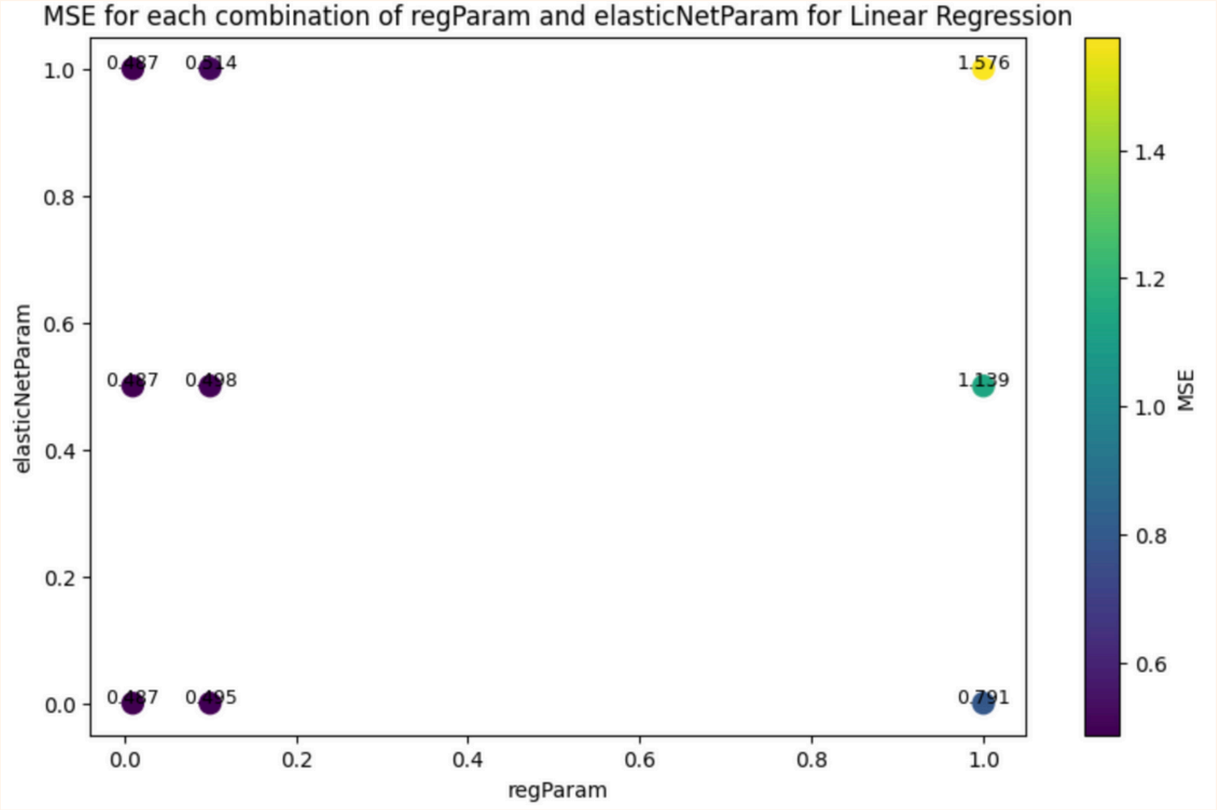
Decision tree regression predicts a target variable by splitting the data into branches based on feature values.

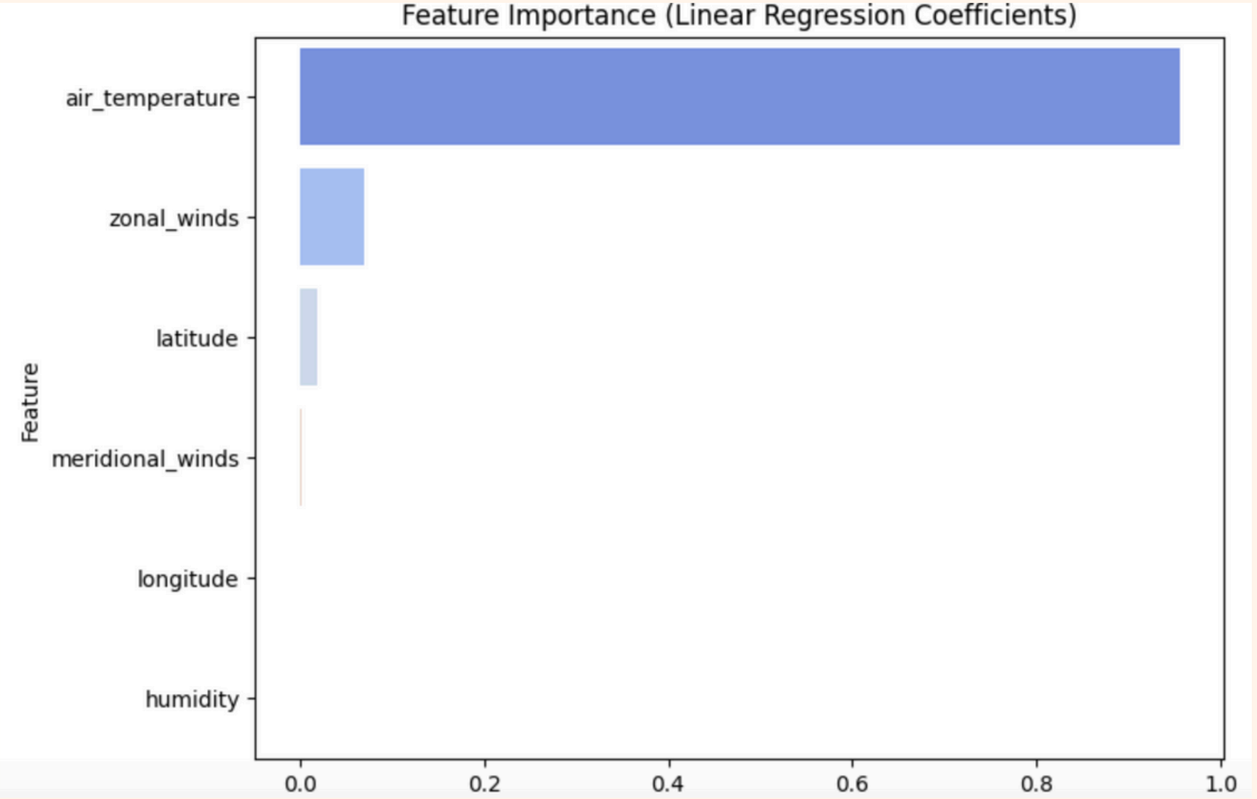
```
rf_paramGrid = ParamGridBuilder() \
    .addGrid(rf.numTrees, [10, 20]) \
    .addGrid(rf.maxDepth, [5, 10]) \
    .addGrid(rf.maxBins, [16, 32]) \
    .build()

rf_crossval = CrossValidator(estimator=rf,
                             estimatorParamMaps=rf_paramGrid,
                             evaluator=evaluator,
                             numFolds=3)
```

METHODOLOGY:

RESULT - LINEAR REGRESSION



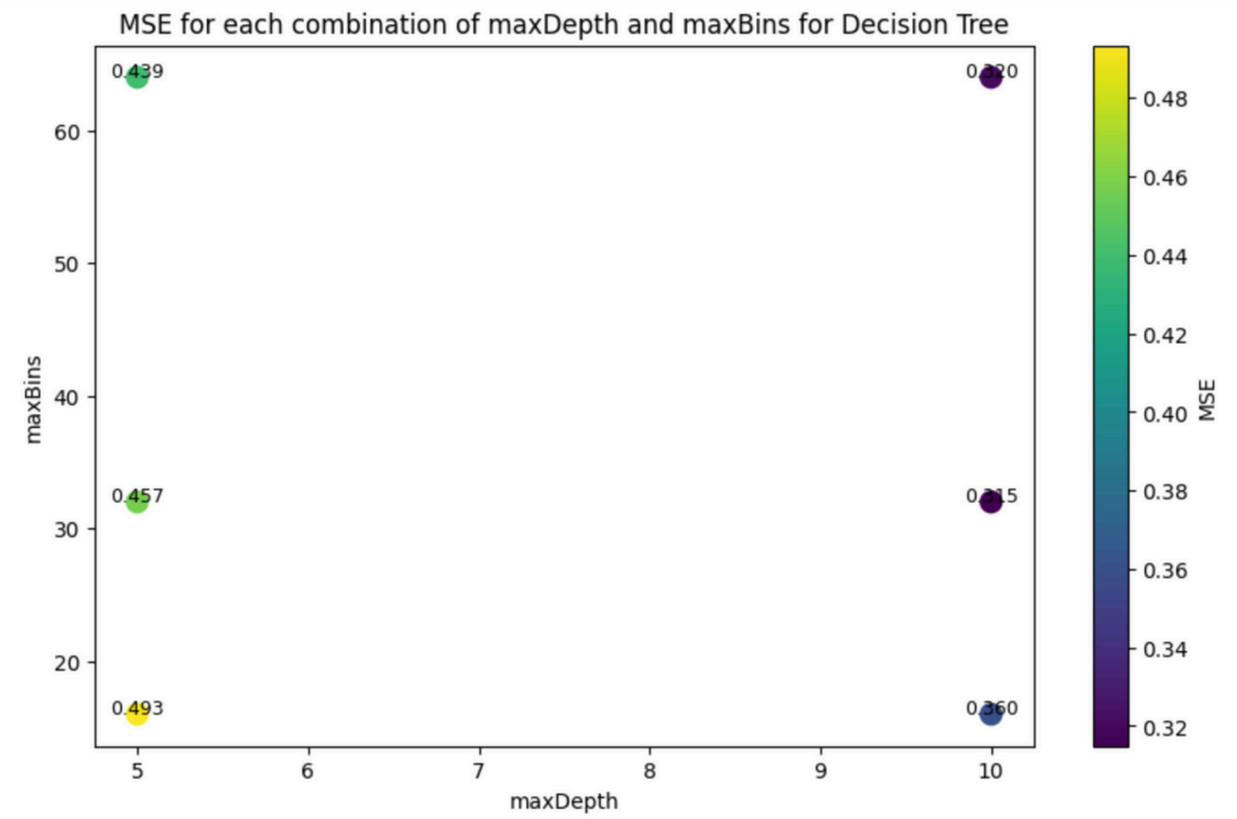


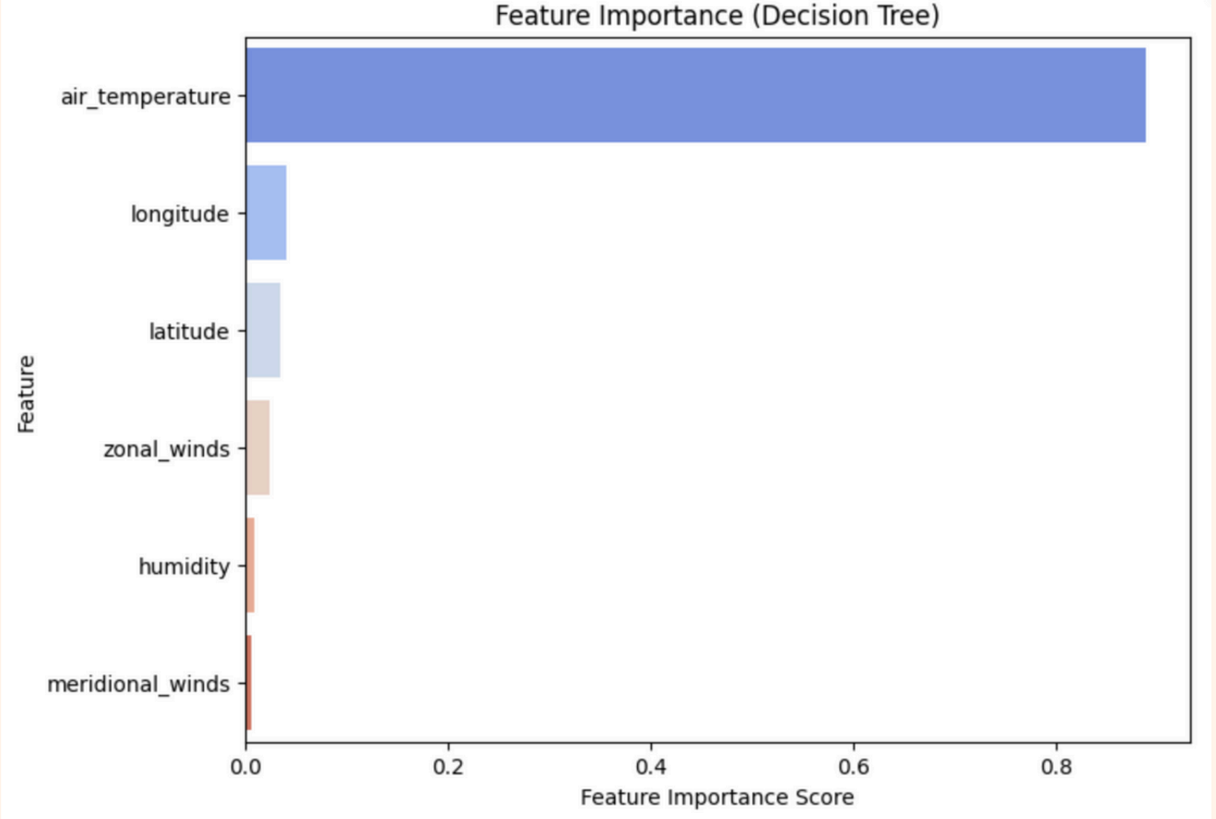

```
Best regParam: 0.01
Best elasticNetParam: 0.0
```

```
Linear Regression MSE: 0.49
Linear Regression MAE: 0.5
Linear Regression R²: 0.85
```

METHODOLOGY:

RESULT - DECISION TREE



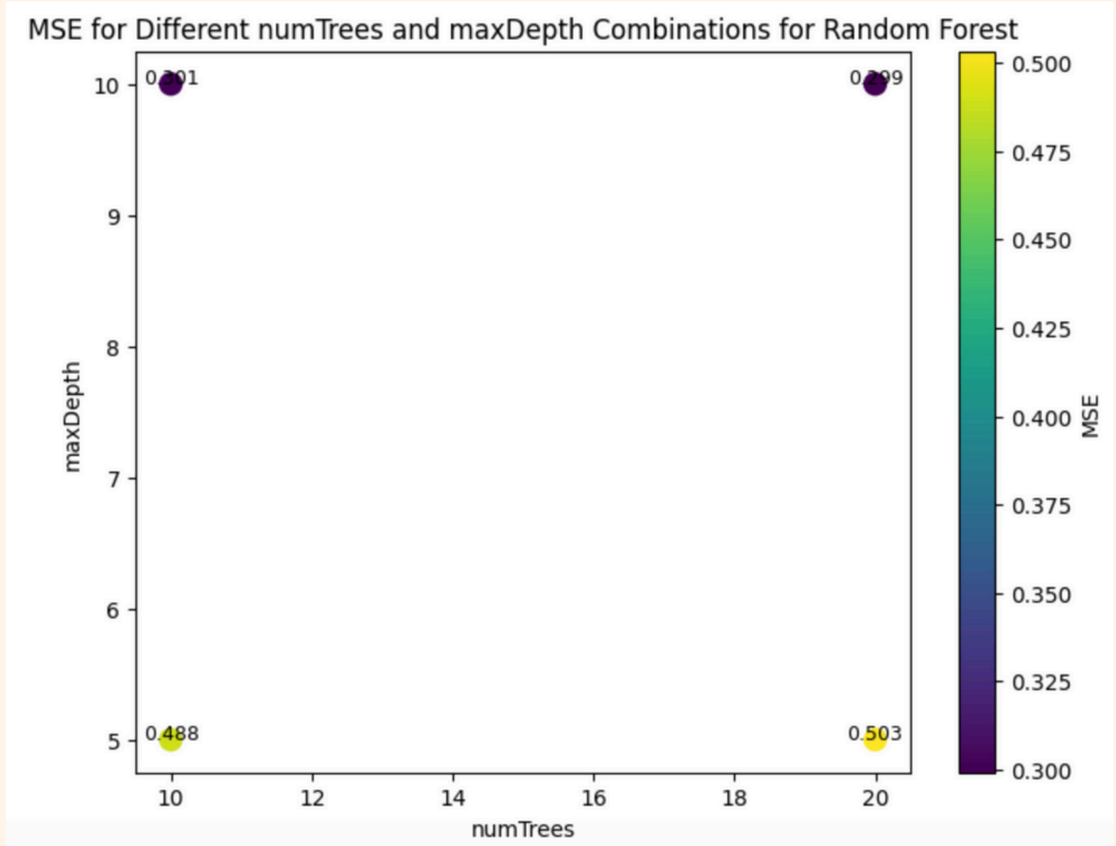


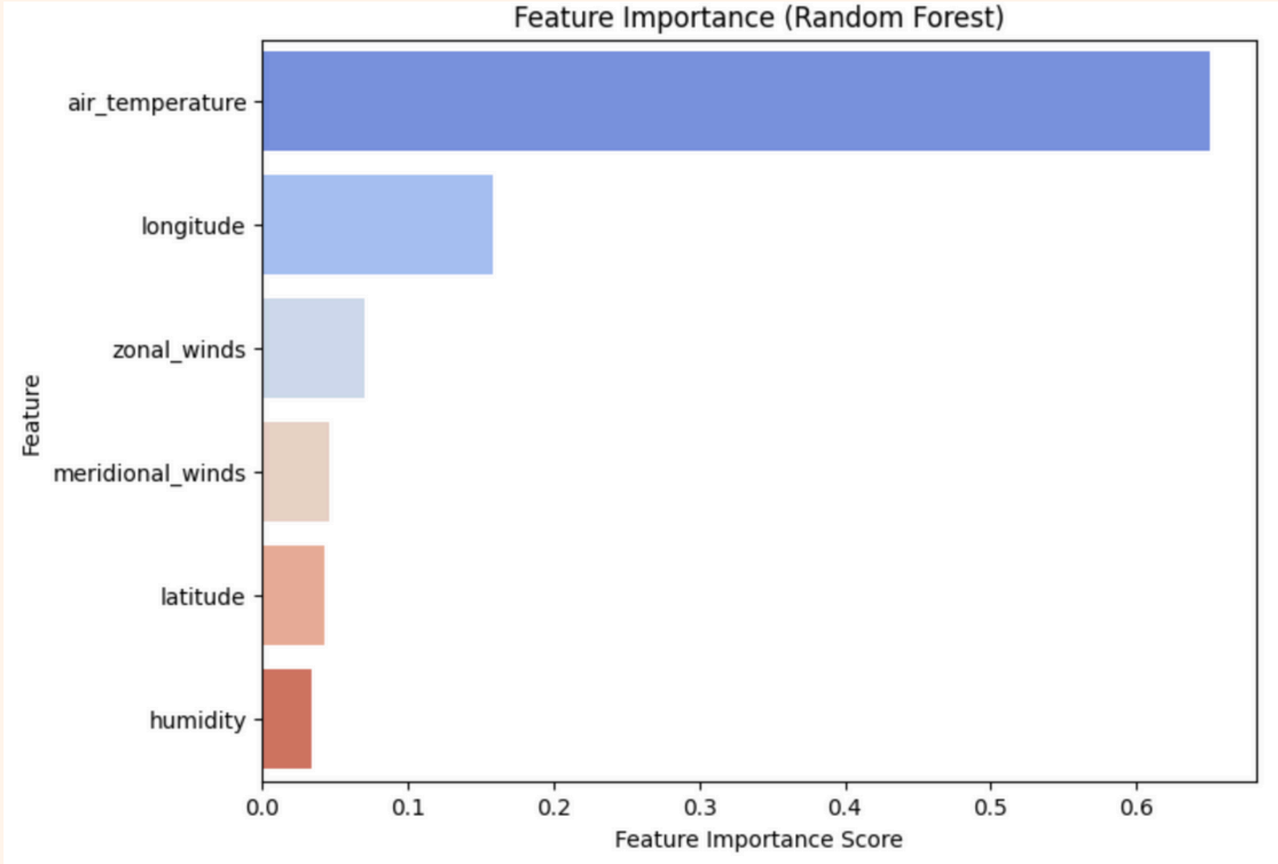

```
Best maxDepth: 10
Best maxBins: 64
```

```
Decision Tree MSE: 0.31
Decision Tree MAE: 0.39
Decision Tree R²: 0.9
```

METHODOLOGY:

RESULT - RANDOM FOREST






```
Best numTrees: 20
Best maxDepth: 10
```

```
Random Forest MSE: 0.29
Random Forest MAE: 0.39
Random Forest R²: 0.91
```

PROJECT WORK IN BIG DATA ANALYTICS:

CLUSTERING WEATHER CONDITIONS

Objectives

Identify distinct patterns or groupings in the atmospheric and oceanic variables to better understand their relationships and variations during El Niño events

METHODOLOGY:

TRAINING AND EVALUATION

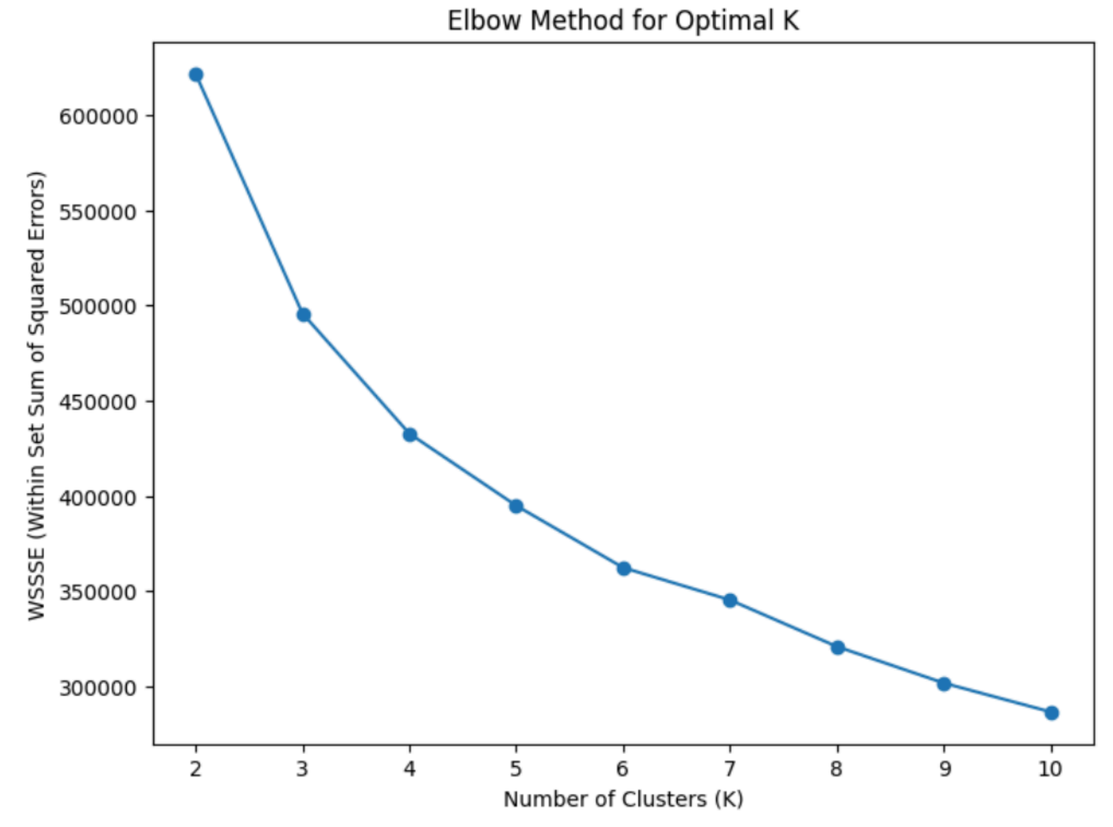
Vector Assembler

PySpark transformer that consolidates multiple feature columns into a single vector column

Standard Scaler

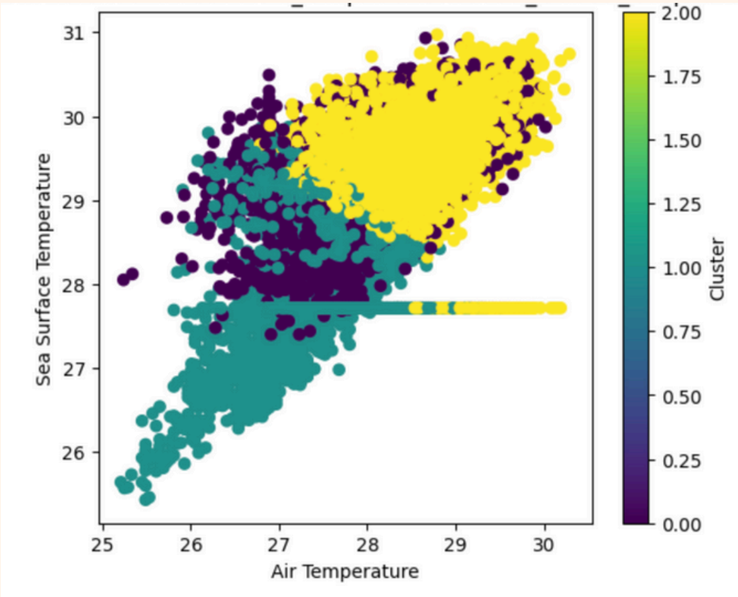
K-means relies on Euclidean distance to measure the similarity between points. If the data have different ranges, the larger-scaled features will dominate the distance computation, which can bias the clustering results.

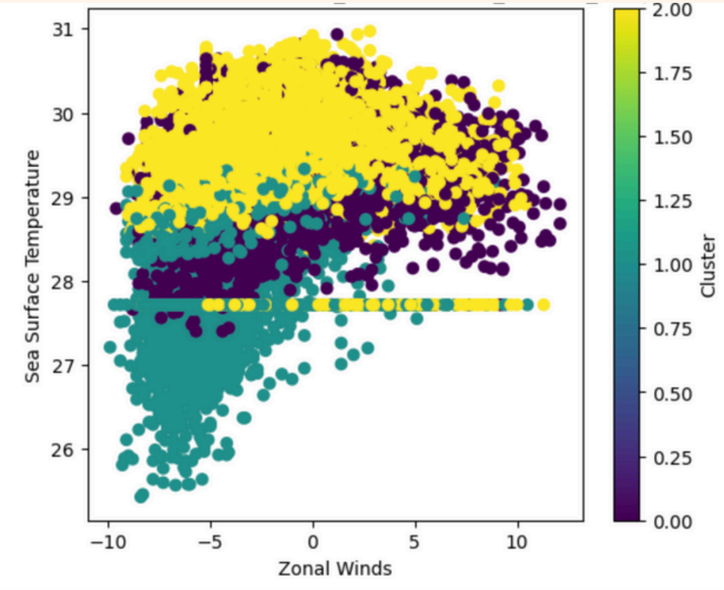
Finding Optimal K

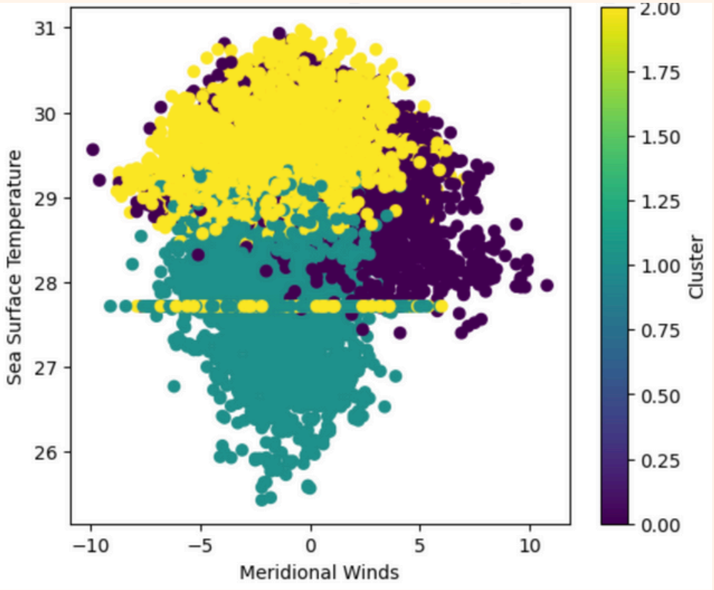


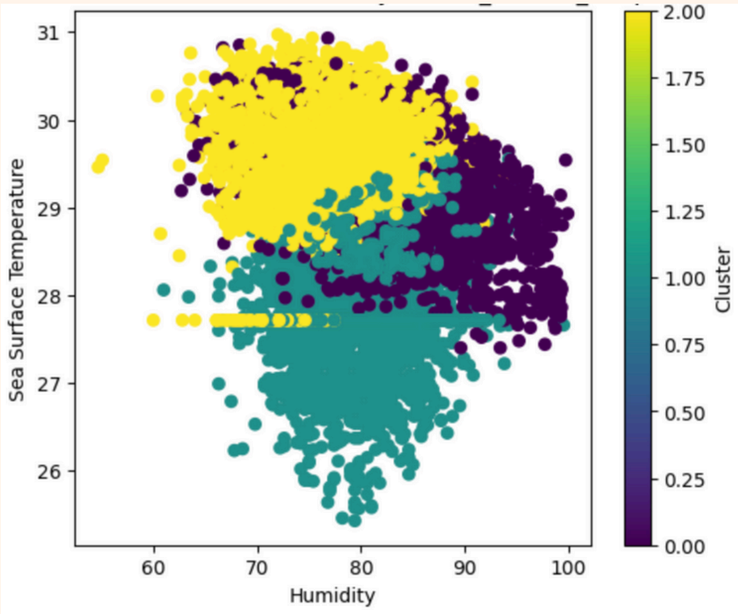
METHODOLOGY:

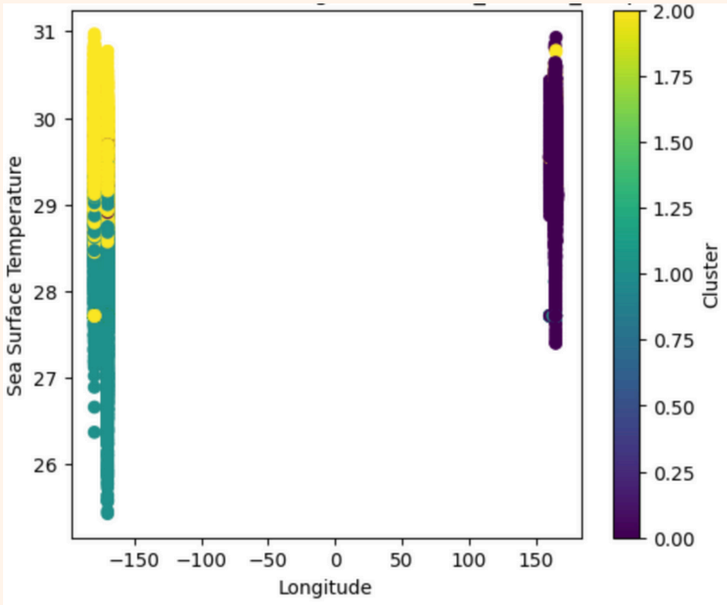
RESULT - WEST PACIFIC





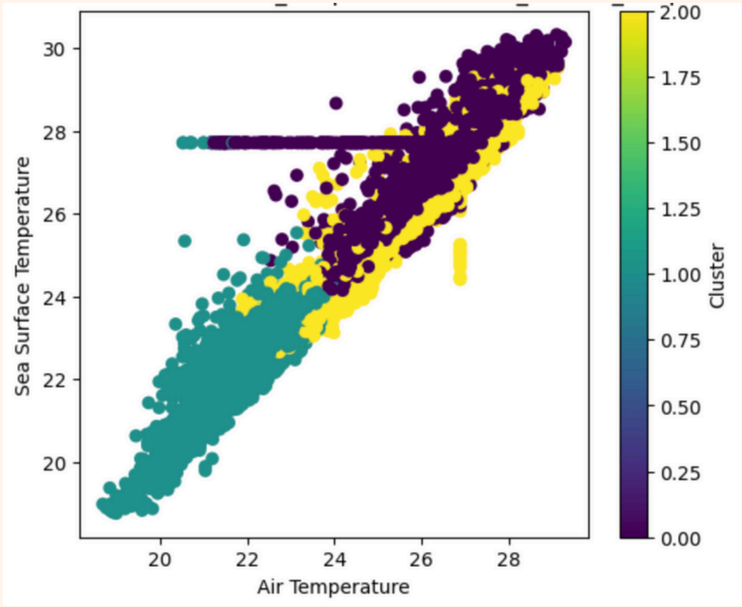


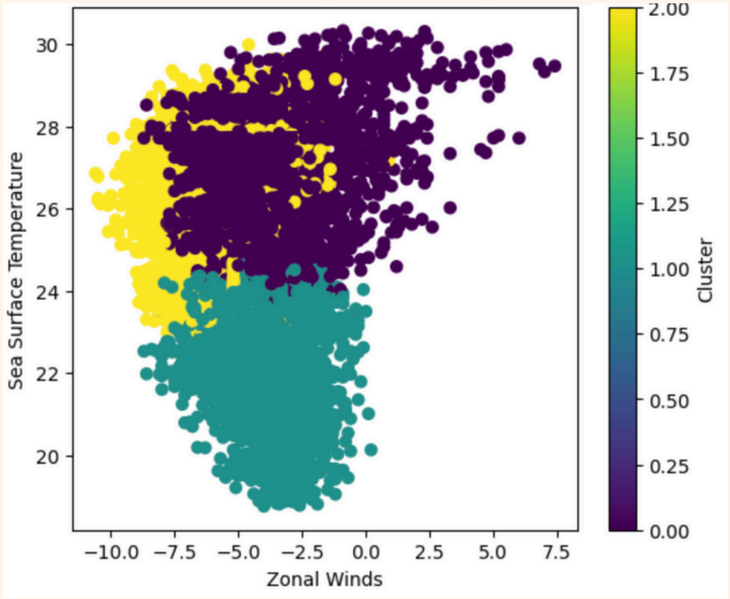


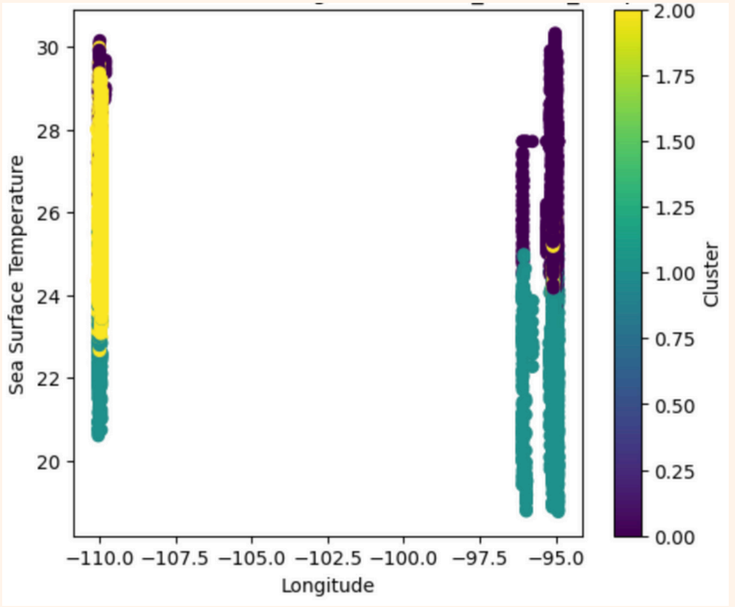


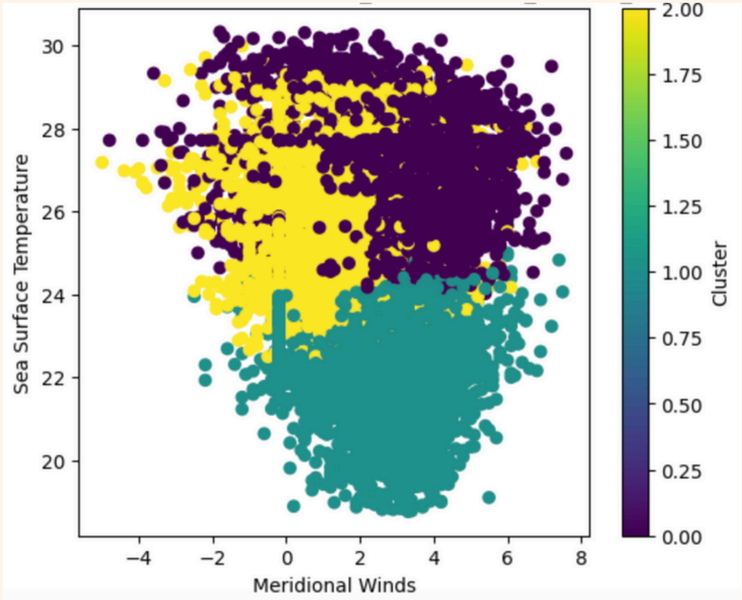
METHODOLOGY:

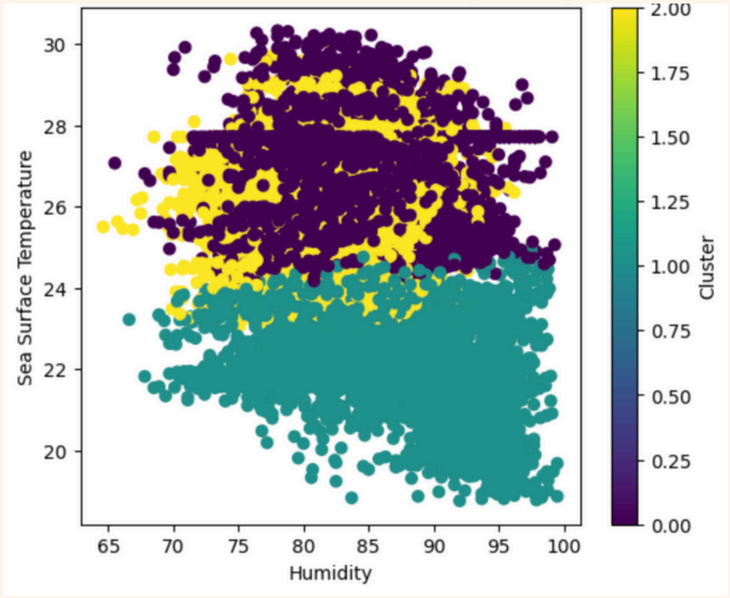
RESULT - EAST PACIFIC











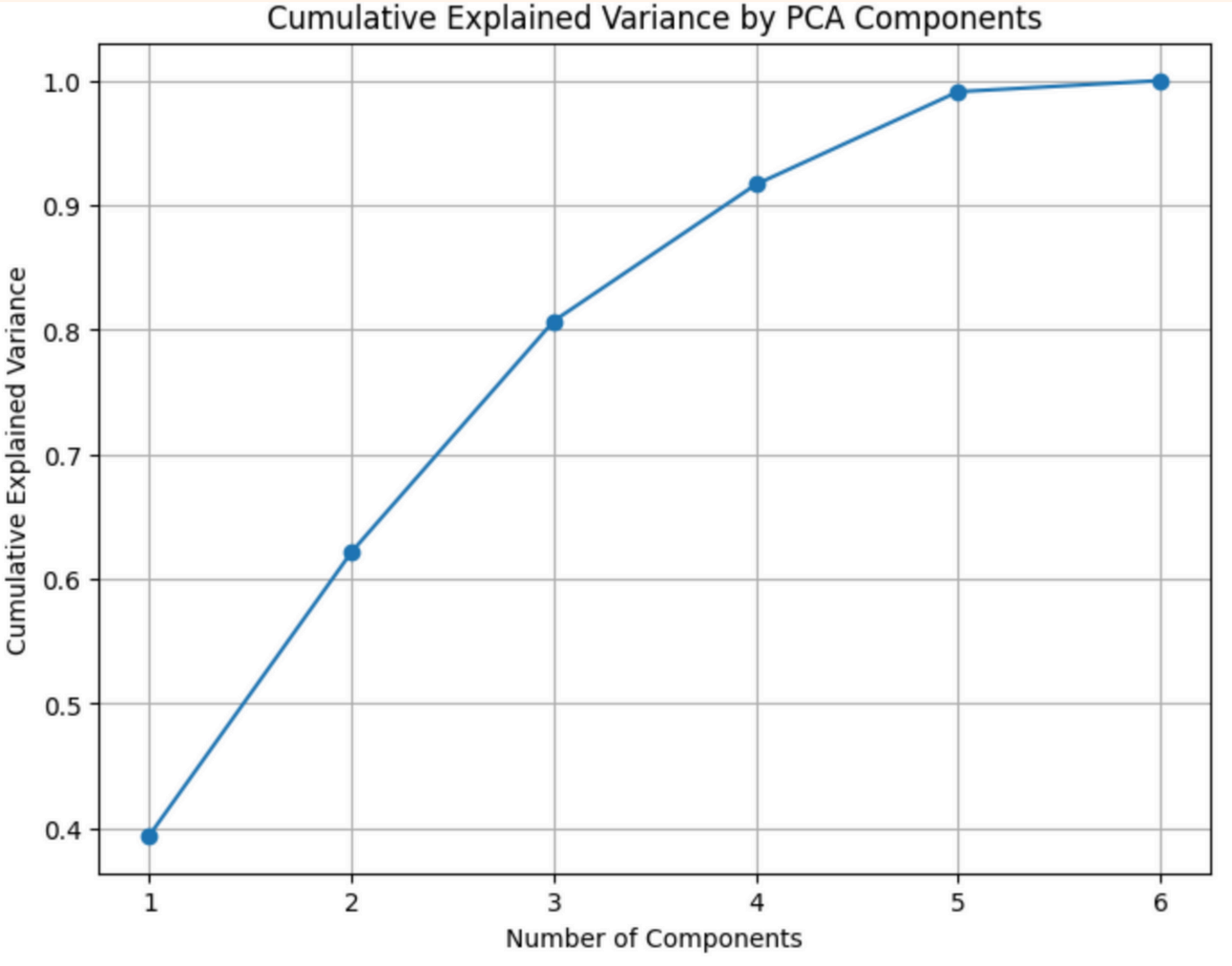
From the clustering task, we can notice that:
1. Features like winds and air temperature shows clearer clusters on East Pacific Region
2. Features like humidty shows clearer clusters on West Pacific Region

However overall, we can not say that this clustering task performs well.
We will use another techniques: PCA + Kmeans Clustering.

METHODOLOGY:

PCA + K MEANS CLUSTERING

Finding Optimal Number of Components



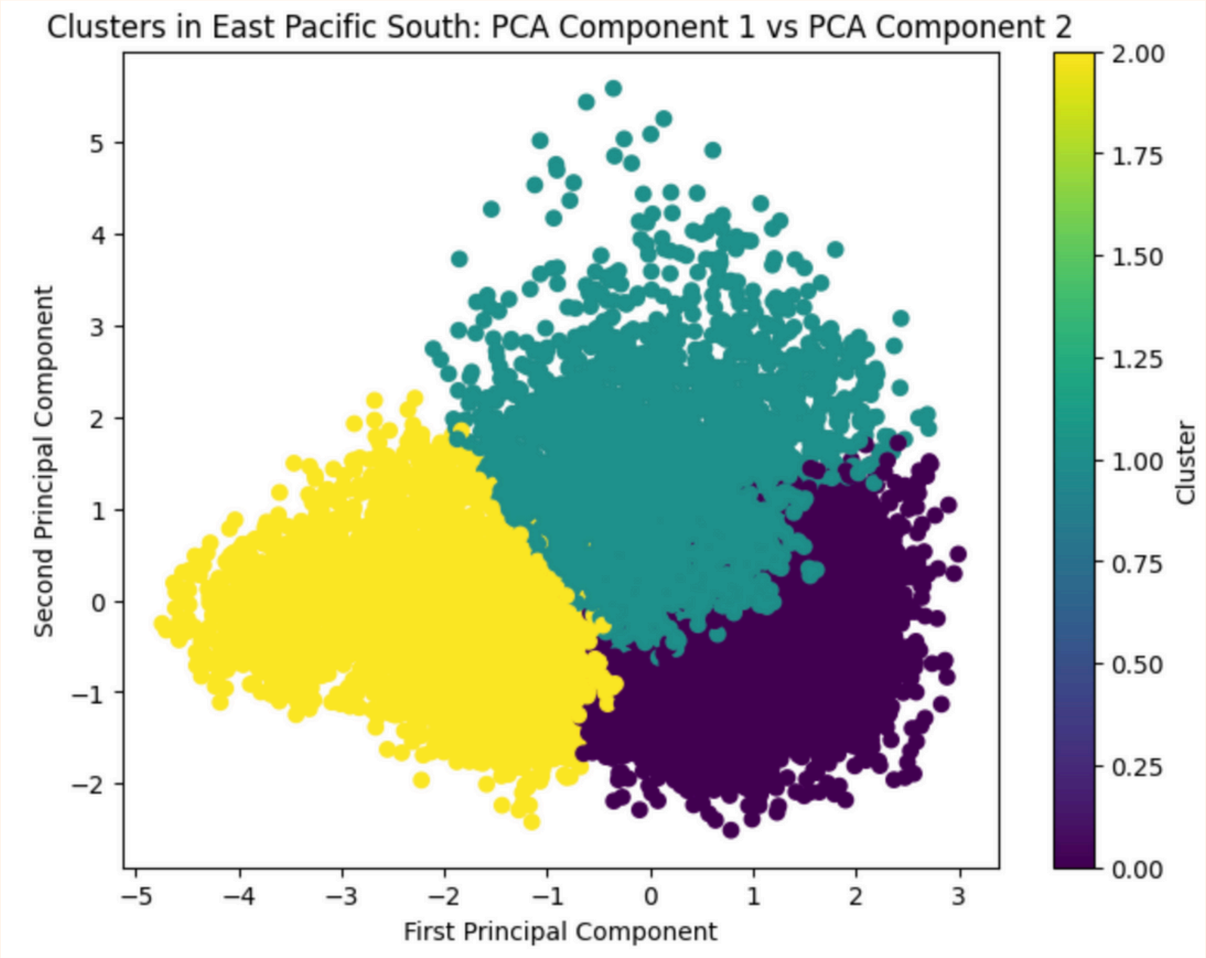
The graph shows the amount of variance captured (on the y-axis) depending on the number of components we include (the x-axis).

A rule of thumb is to preserve around 80 % of the variance. So, in this instance, we decide to keep 3 components.

METHODOLOGY:

PCA + K MEANS CLUSTERING

Result



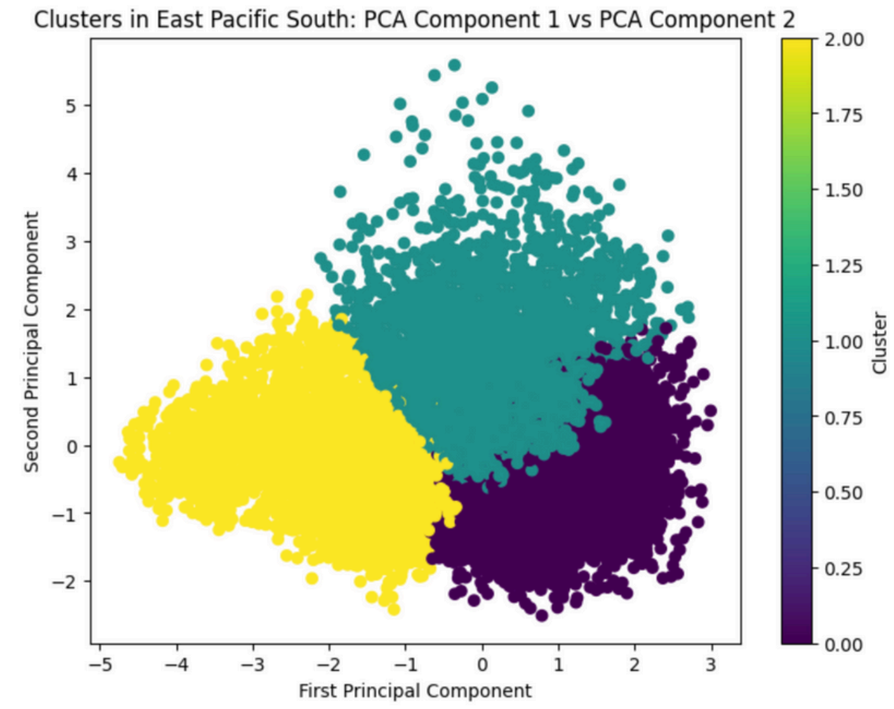
	PCA_1	PCA_2	PCA_3	PCA_4	PCA_5	PCA_6	Cluster
0	1.070251	1.174929	-0.499982	-0.289609	-1.373350	0.088023	1
1	1.150108	0.853172	-0.864760	0.118956	-0.695793	0.170664	0
2	1.064986	0.818959	-0.490686	0.466114	-0.691048	0.219083	0
3	0.877695	1.089315	-1.069247	0.412525	-0.732231	0.218589	0
4	1.045185	0.717737	-0.727375	0.840857	-0.287044	0.217989	0
...	...	...	...	...	...	...	...
11248	0.855579	0.495089	1.909592	-0.506499	0.774273	-0.046611	1
11249	0.778311	0.586305	2.094552	-0.063377	0.850741	0.074269	1
11250	0.178317	0.870913	1.766615	0.432071	0.790001	-0.110670	1
11251	0.182002	0.997058	2.596122	-0.002231	0.030302	-0.005775	1
11252	0.872012	0.853818	2.241165	0.305727	0.913637	0.033899	1

11253 rows × 7 columns

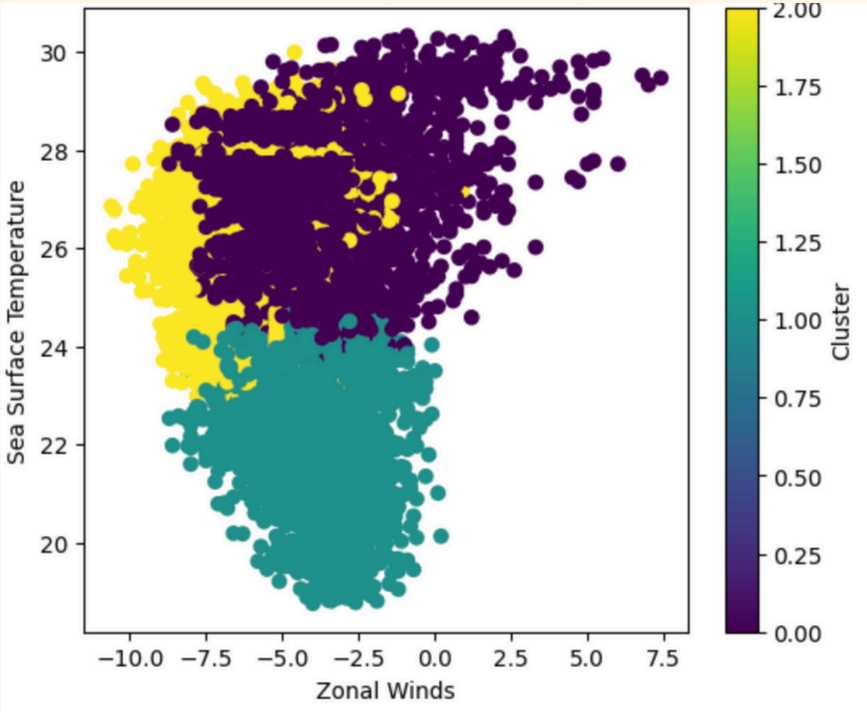
METHODOLOGY:

COMPARISON - WITH AND WITHOUT PCA

With PCA



Without PCA



Without PCA, the data points are mixed and hard to distinguish. However, after performing PCA, the separation becomes more apparent, and the clusters are well-defined. This demonstrates one of the primary goals of PCA: to reduce the dimensionality of the data by combining the original features into fewer, more meaningful components. PCA makes it easier for the K-means algorithm to group similar data points together.

CONCLUSION

Most important features

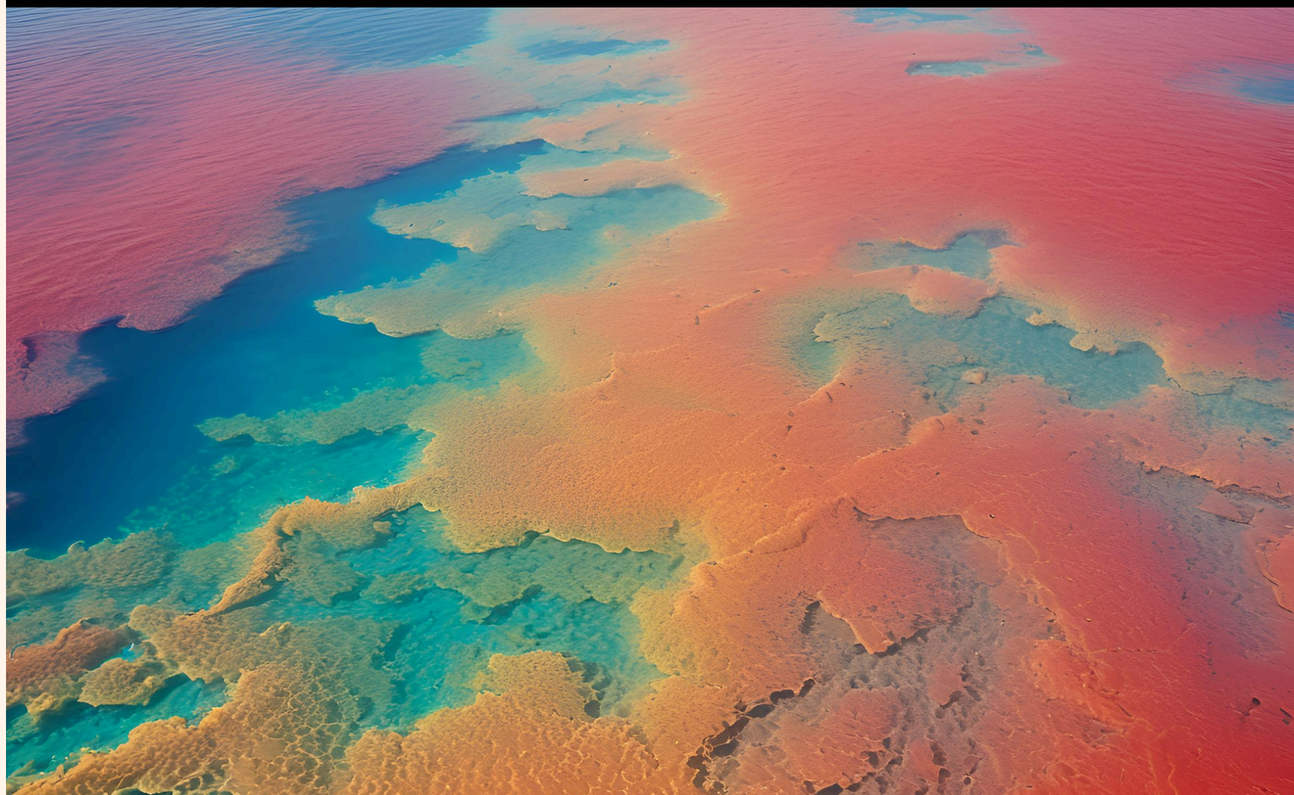
AirTemperature

Best Regression Algorithm

Random Forest with numTrees 20 and maxdepth 10 which produce MSE = 0.3

Best k for Clustering

k = 3 and number of components of PCA = 3 which produces sillhoutte score: 0.7 for East Pacific Region




THANK
YOU