Answers and Decisions Document

In this document we will answer the how of the project and explain the why of the decisions taken.

We have two datasets:

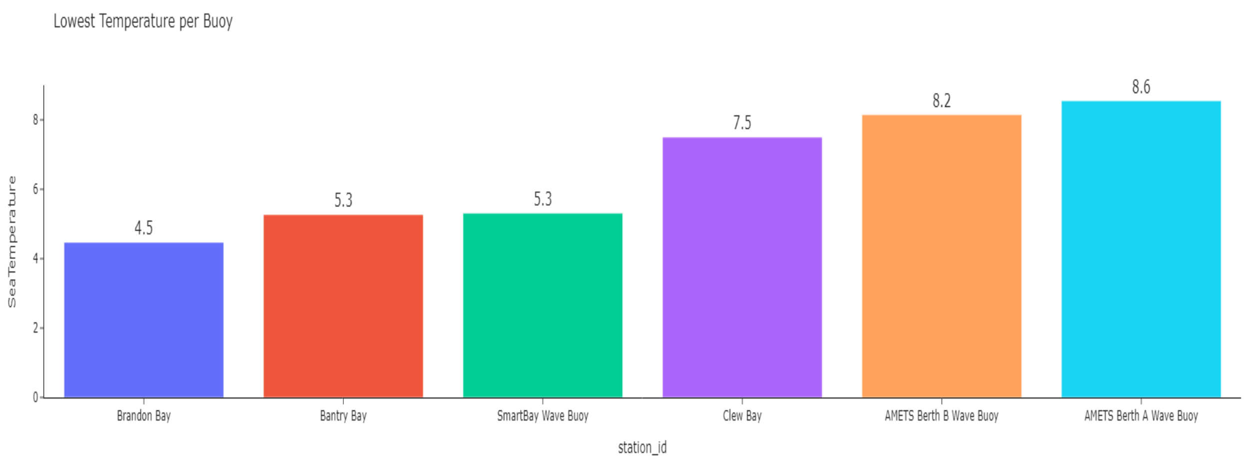
Dataset	***Description***
Tides	Measures about **Tides** collected by several bouys at the Ireland Sea
Waves	Measures about **Waves** collected by several bouys at the Ireland Sea

The data were collected between 2021-09-16 00:01 and 2022-09-16 23:59.

The first question of the project it is:

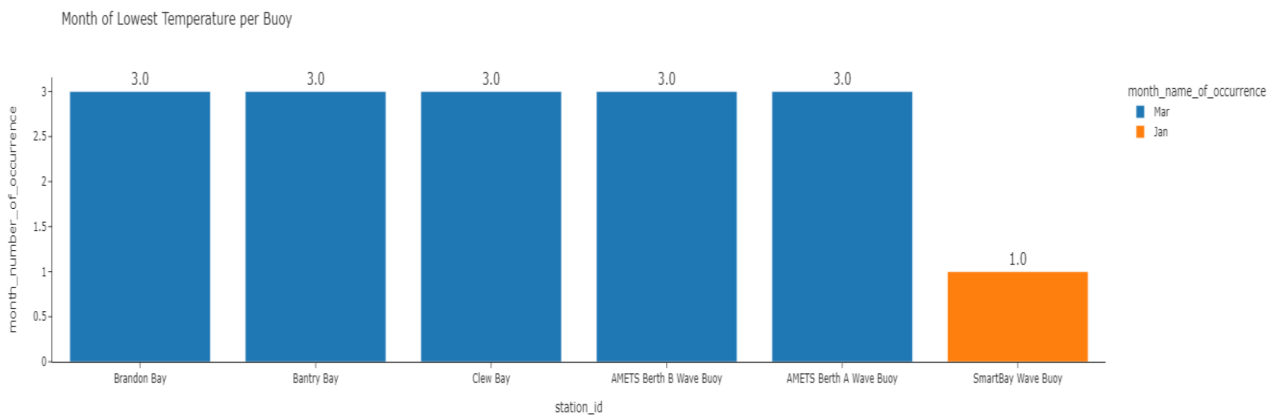
1. **What is the lowest temperature of each one of the Buoys?**

To answer this question, we have to analyze the waves dataset since the buoys are used to measure Waves data.



We can see that the lowest temperature registered was 4.5, located in the Brandon Bay, and the highest temperature was 8.6, registered in the AMETS Berth A Wave Buoy.

1.1 Which month it usually occurs?



We see that for the majority of the buoys, the lowest temperature occurs in March, except for the SmarBay Wave Buoy that occurs in January. March it is the final month of the Winter in the north hemisfery.

2. Where (lat/long) do we have the biggest water level?

When we talk about water level, we have to analyze the Tides dataset.

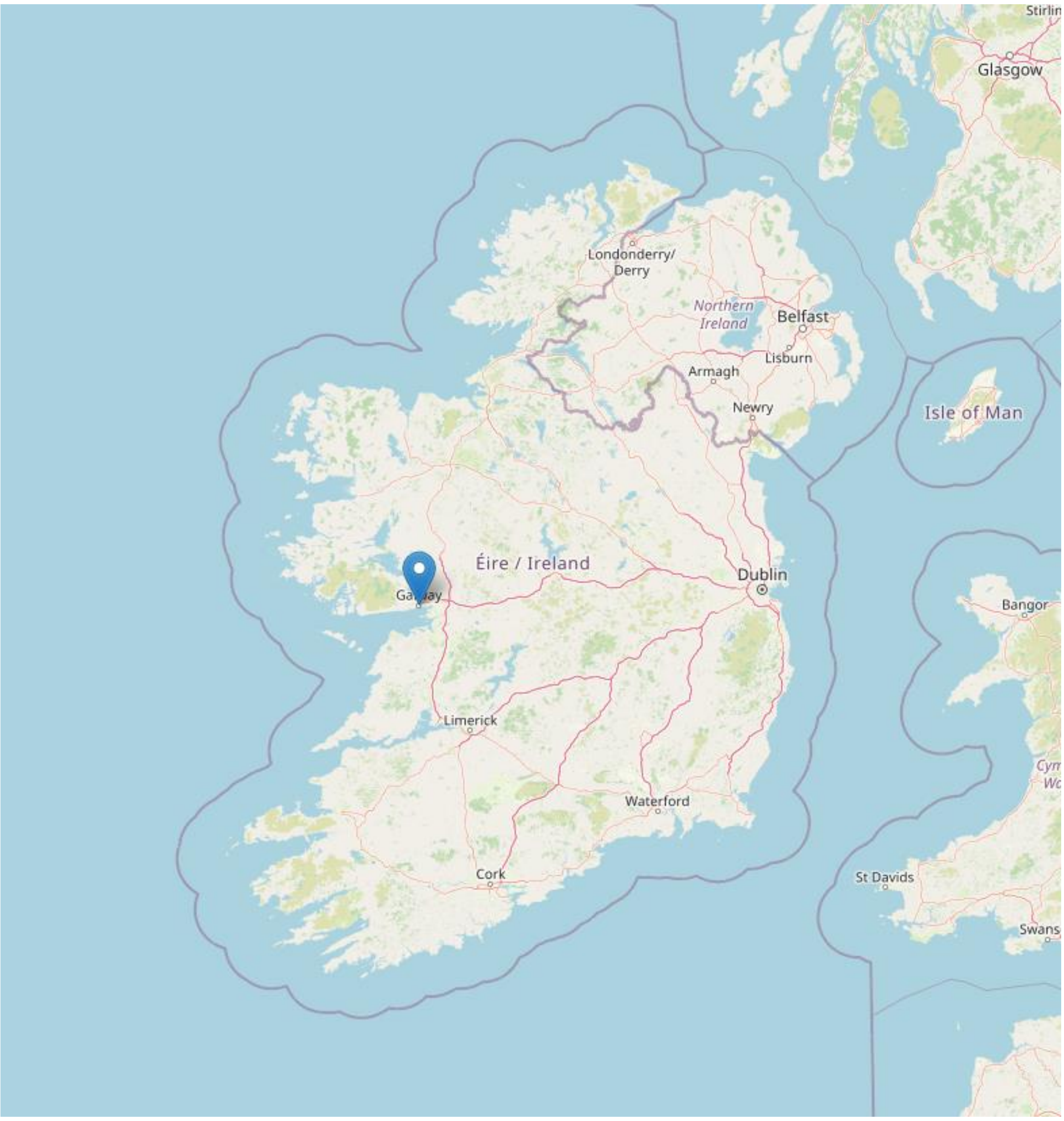
Since there are two water level on the dataset, I did a little research to know wich one to use.

Lowest astronomical tide (LAT) is defined as the lowest tide level which can be predicted to occur under average meteorological conditions and under any combination of astronomical conditions.

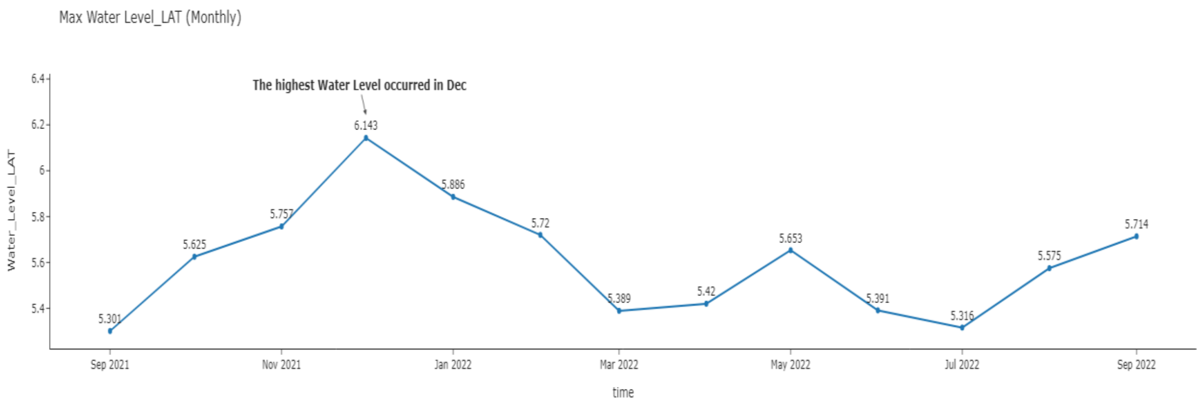
An ordnance datum or OD is a vertical datum used by an ordnance survey as the basis for deriving altitudes on maps. OD usually it is relative to the MSL (Mean Sea Level).

That said, I decided to use LAT because it it more widely used.

The highest water level was detected in the 53.269, -9.048 coordinates. More accurately, on 53°16'08.4"N+9°02'52.8"W, located on the Galway Port.

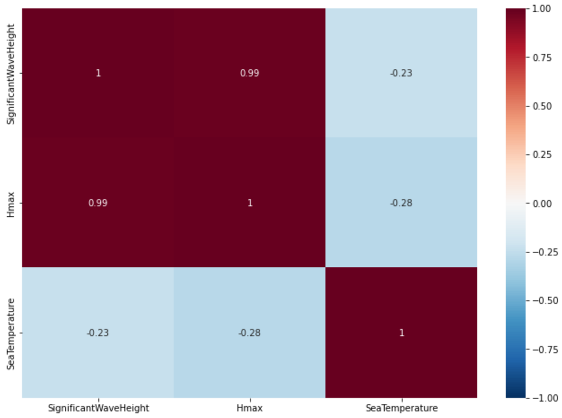


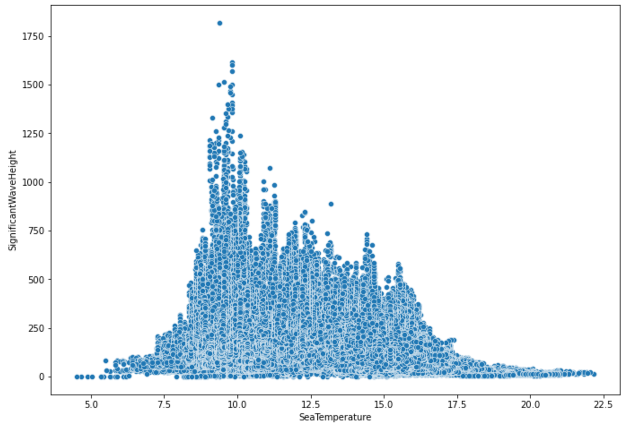
2.1- Which usually month it occurs?



The highest water level was recorded on December, 2021.

3. How the Wave Lengths correlates with Sea Temperature?





The wave height has a weak and negative correlation with Sea Temperature. That means that higher the Sea Temperature, lower the wave height.

3.1 It is possible to predict with accuracy the Wave Length, based on the Sea Temperature and the Buoy location?

To answer this question, we have to dig a little deep on the dataset.

First, let's see if the means of sea temperature and the wave height differ between buoys.

	station_id	SignificantWaveHeight								SeaTemperature							
		count	mean	std	min	25%	50%	75%	max	count	mean	std	min	25%	50%	75%	max
0	AMETS Berth A Wave Buoy	16812.0	316.699798	185.696164	2.0	181.0	274.0	403.0	1617.0	16812.0	12.413886	2.292191	8.55	10.10	12.30	14.3500	17.45
1	AMETS Berth B Wave Buoy	16569.0	283.064518	160.940415	2.0	166.0	247.0	362.0	1819.0	16569.0	11.960142	2.232190	8.15	9.65	12.05	14.0000	16.55
2	Bantry Bay	15706.0	60.362537	41.346492	4.0	30.0	51.0	81.0	486.0	94264.0	12.584481	3.292467	5.27	9.57	12.02	15.0300	22.23
3	Brandon Bay	7212.0	173.526900	91.102908	0.0	111.0	160.0	216.0	628.0	43319.0	11.704346	2.201066	4.47	9.74	10.90	13.4400	16.50
4	Clew Bay	9666.0	119.018105	70.116461	15.0	64.0	105.0	160.0	483.0	9666.0	11.202338	2.608078	7.50	9.00	10.00	13.0875	18.40
5	SmartBay Wave Buoy	11717.0	82.665358	49.994963	6.0	43.0	74.0	108.0	300.0	70362.0	12.479693	3.799434	5.31	8.70	12.44	16.5000	22.36

We can see that the mean and standard deviation of the Wave Heights seem to differ between buoys, although Sea Temperature don't.

Let's investigate with some statistics tests if the buoy location it is statically significant.

ANOVA

Since we are analyzing the effect of a categorical variable (Buoy) in a continuous variable (Wave Height), we have to perform as ANOVA test.

An ANOVA test is a type of statistical test used to determine if there is a statistically significant difference between two or more categorical groups by testing for differences of means using variance.

One Way ANOVA

The one-way ANOVA tests the null hypothesis that two or more groups have the same population mean. The test is applied to samples from two or more groups, possibly with differing sizes.

Our groups are the Buoys Location, and our target it is the Wave Heights

The ANOVA test has important assumptions that must be satisfied in order for the associated p-value to be valid.

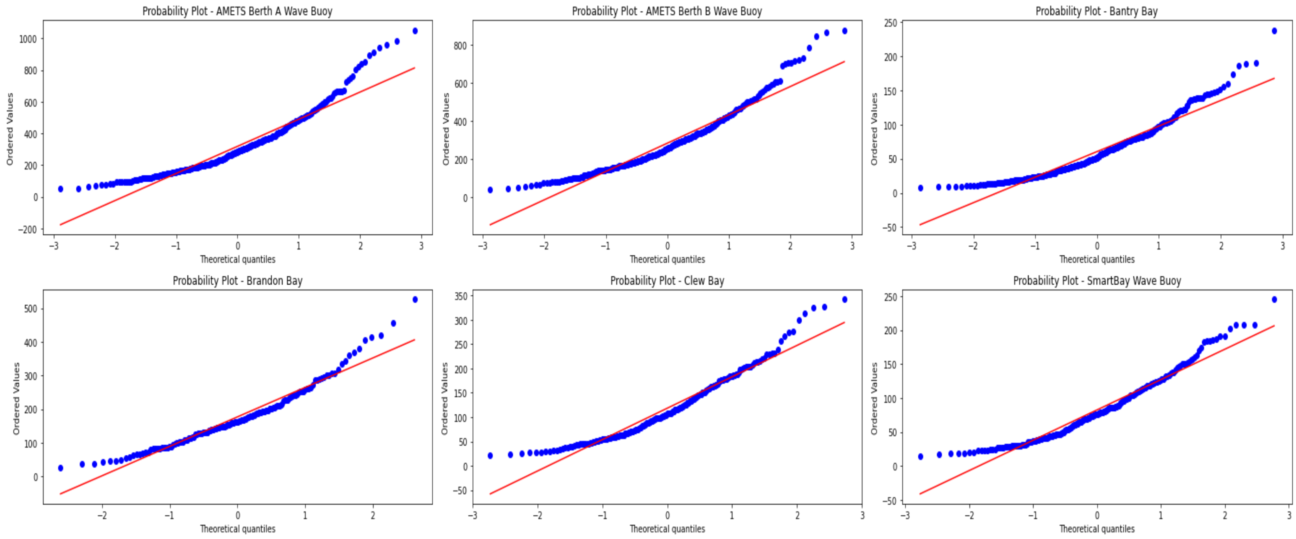
- The samples are independent.

- Each sample is from a normally distributed population.

- The population standard deviations of the groups are all equal. This property is known as homoscedasticity.

Let’s test it!

Checking data normality for each Buoy



There are evidences that the data is not normally distributed, the QQ-Plots shows us that the data are skewed.

Check if standard deviations of the groups are all equal

__Levene Test__

The Levene test tests the null hypothesis that all input samples are from populations with equal variances. Levene’s test is an alternative to Bartlett’s test bartlett in the case where there are significant deviations from normality.

H0: μ_1 = μ_2 ("there is no difference in the variance of wave height between buoys")

H1: μ_1 ≠ μ_2$ ("there is a difference in the variance of wave height between buoys")

$H_0 : \mu_1 = \mu_2$ ("there is no difference in the variance of wave height between buoys")

$H_1 : \mu_1 \neq \mu_2$ ("there is a difference in the variance of wave height between buoys")

```
scipy.stats.levene(
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'AMETS Berth A Wave Buoy')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'AMETS Berth B Wave Buoy')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'Bantry Bay')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'Brandon Bay')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'Clew Bay')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'SmartBay Wave Buoy')][target_column],
center='mean')
```

```
LeveneResult(statistic=114.14031010181242, pvalue=7.873188829913835e-104)
```

Since the P-Value it is lower than 0.05, there is a difference in the variance of wave height between buoys.

We didn't fill 2 of the 3 assumptions to ANOVA, that means that our results may not be fully trusted, but just out of curiosity, let's test the result of the test.

Anova Hyptohesis

- $H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4 = \mu_5 = \mu_6$ (the six population means are equal)
- H_1 : At least one of the means differ

```
scipy.stats.f_oneway(
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'AMETS Berth A Wave Buoy')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'AMETS Berth B Wave Buoy')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'Bantry Bay')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'Brandon Bay')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'Clew Bay')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'SmartBay Wave Buoy')][target_column],
)
```

```
F_onewayResult(statistic=267.3609349154022, pvalue=8.928212798194785e-210)
```

Since the P-Value it is lower than 0.05, we reject the null hypothesis as there is significant evidence that at least one of the means differ.

Kruskal-Wallis H-test

If one of ANOVA assumptions are not true for a given set of data, it may still be possible to use the Kruskal-Wallis H-test (scipy.stats.kruskal) although with some loss of power.

The Kruskal-Wallis H-test tests the null hypothesis that the population median of all of the groups are equal. It is a non-parametric version of ANOVA. The test works on 2 or more independent samples, which may have different sizes. Note that rejecting the null hypothesis does not indicate which of the groups differs. Post hoc comparisons between groups are required to determine which groups are different.

```
scipy.stats.kruskal(
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'AMETS Berth A Wave Buoy')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'AMETS Berth B Wave Buoy')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'Bantry Bay')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'Brandon Bay')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'Clew Bay')][target_column],
    df_waves_per_station_day_mean.loc[(df_waves_per_station_day_mean['station_id'] == 'SmartBay Wave Buoy')][target_column],
    nan_policy='omit'
)
```

```
KruskalResult(statistic=993.0160226868268, pvalue=1.953777160551531e-212)
```

Since the P-Value it is lower than 0.05, we reject the null hypothesis as there is significant evidence that at least one of the means differ.

<u>Supervised Model</u>

Since one of the means differs from others, it worths a shot creating a basic model to see how it performs.

Our target variable it's Wave Height (cm), that means that we need a supervised machine learning model and our target it is a continuous variable, a regression problem.

I've chose to use RMSE as our metric because it is more sensible to outliers than MAE, so it gives us a wider comprehension if that is affecting our model.

For validation, we will use K-Fold Cross Validation. That means that the data will be divided by K groups of samples, called folds. Then, in every iteration of K, the data will be trained in K-1 and tested in the rest.

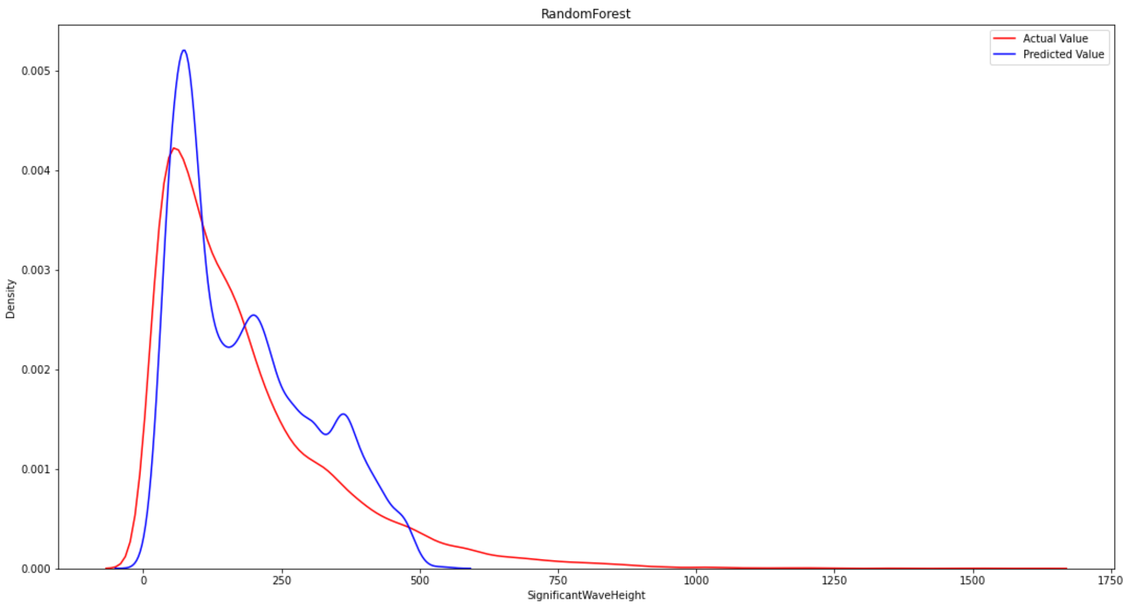
For the model, I chose to work with random forest, that uses bagging technique and usually performs well.

I splitted the data into train and validation, in the ratio of 0.75/0.25, respectively.

For the data transformation, I performed a One Hot Encoding for categorical variables and a Standard Scaler (mean=0, std=1) for the continuous variables.

Our results were:

	cv_rmse	cv_std	rmse	mae	r2
0	107.593209	1.627879	106.996486	70.109342	0.562239



<u>Conclusion</u>

Our tests shows that there is significant evidence that at least one the means for Wave Height values differ between Buoys.

This was an indicator that the Buoy location may influence the Wave Height, but we can't be sure. Since the Sea Temperature is not a strong feature for predicting the Wave Height, I thought it worth creating a basic model to see how it performs.

With RandomForest we obtained a RMSE of 107.59 with cross-validation, and our r2 score it is 0.56 which indicates that our features may not explain the target very well.

The model didn't perform very well**, indicating that these 2 features are not enough to predict the Wave Height accurately**, but with some adjustments it may improve.

4. Bonus Answers

4.1 Build a Time Series model that can predict the sea temperature throughout the year.

To start building the model, we have to know the data, so it is necessary a exploratory data analysis.

```
df_waves.head()
```

[7] ✓ ✓ 0.1s

	longitude	latitude	time	station_id	PeakPeriod	PeakDirection	UpcrossPeriod	SignificantWaveHeight	Hmax	SeaTemperature	MeanCurSpeed	MeanCurDirTo
0	-9.262278	53.228333	2021-09-16 00:05:00	SmartBay Wave Buoy	NaN	NaN	NaN	NaN	NaN	16.72	NaN	NaN
1	-9.262278	53.228333	2021-09-16 00:10:00	SmartBay Wave Buoy	NaN	NaN	NaN	NaN	NaN	16.72	0.108	23.912088
2	-9.262278	53.228333	2021-09-16 00:15:00	SmartBay Wave Buoy	NaN	NaN	NaN	NaN	NaN	16.72	NaN	NaN
3	-9.262278	53.228333	2021-09-16 00:20:00	SmartBay Wave Buoy	NaN	NaN	NaN	NaN	NaN	16.72	0.138	29.010988
4	-9.262278	53.228333	2021-09-16 00:25:00	SmartBay Wave Buoy	NaN	NaN	NaN	NaN	NaN	16.72	NaN	NaN

```
df_waves.tail()
```

[8] ✓ 0.1s

	longitude	latitude	time	station_id	PeakPeriod	PeakDirection	UpcrossPeriod	SignificantWaveHeight	Hmax	SeaTemperature	MeanCurSpeed	MeanCurDirTo
16564	-10.15099	54.2251	2022-09-05 21:32:00	AMETS Berth B Wave Buoy	9.09	233.4	4.819	112.0	NaN	15.2	NaN	NaN
16565	-10.15099	54.2251	2022-09-05 22:02:00	AMETS Berth B Wave Buoy	10.00	232.0	4.762	123.0	NaN	15.2	NaN	NaN
16566	-10.15099	54.2251	2022-09-05 22:32:00	AMETS Berth B Wave Buoy	10.53	229.2	5.000	123.0	NaN	15.2	NaN	NaN
16567	-10.15099	54.2251	2022-09-05 23:02:00	AMETS Berth B Wave Buoy	10.53	229.2	5.714	140.0	NaN	15.2	NaN	NaN
16568	-10.15099	54.2251	2022-09-05 23:32:00	AMETS Berth B Wave Buoy	10.53	229.2	5.333	140.0	NaN	15.2	NaN	NaN

We can see that the time interval between measures it is not regular, on the beginning it is collected every 5 minutes, and later every 30 minutes.

```
# check for nulls
df_waves.isnull().sum()
```

✓ 0.1s

```
longitude                      0
latitude                       0
time                           0
station_id                     0
PeakPeriod                173310
PeakDirection             173310
UpcrossPeriod             173310
SignificantWaveHeight     173310
Hmax                      216370
SeaTemperature                 0
MeanCurSpeed              198428
MeanCurDirTo              198431
dtype: int64
```

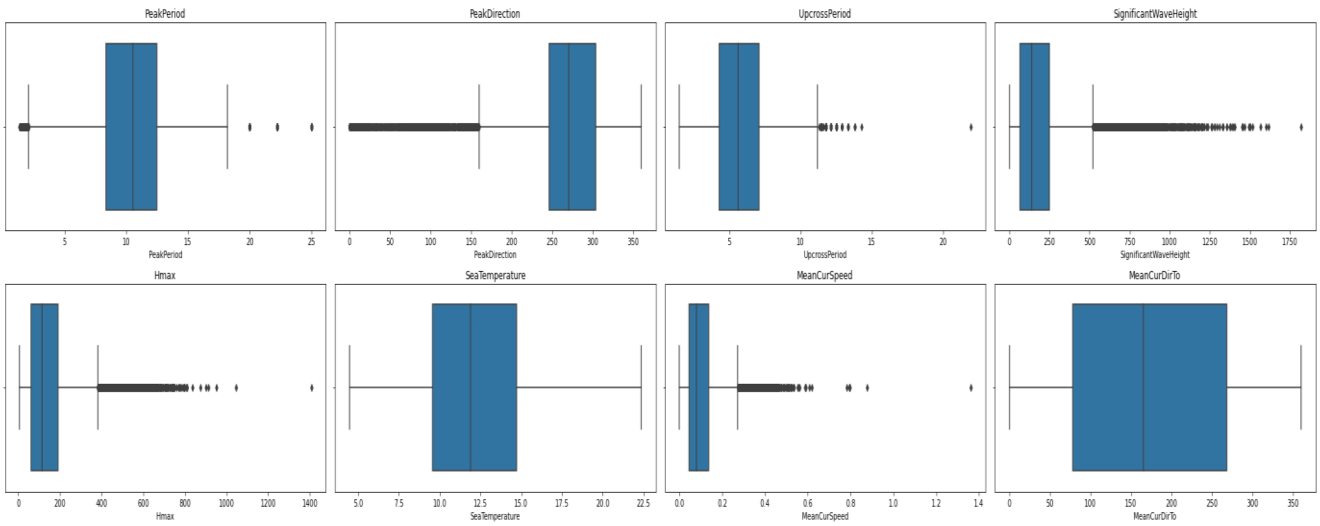
We have a lot of missing data. Judging the name of the column, it was values collected to state aggregations per period.

```
df_waves.describe()
```

✓ 0.1s

	longitude	latitude	PeakPeriod	PeakDirection	UpcrossPeriod	SignificantWaveHeight	Hmax	SeaTemperature	MeanCurSpeed	MeanCurDirTo
count	250992.000000	250992.000000	77682.000000	77682.000000	77682.000000	77682.000000	34622.000000	250992.000000	52564.000000	52561.000000
mean	-9.716230	52.629164	10.258620	267.979840	5.741876	184.508664	143.025995	12.297332	0.101646	173.233028
std	0.345750	0.941411	3.259992	54.317685	1.898708	161.773999	111.588810	3.170566	0.076782	103.141230
min	-10.297370	51.647000	1.350000	0.000000	1.490000	0.000000	6.000000	4.470000	0.000000	0.000000
25%	-10.094833	51.647000	8.330000	246.153840	4.280000	68.000000	62.000000	9.550000	0.046000	77.890110
50%	-9.681000	52.282333	10.530000	270.000000	5.634000	138.000000	114.000000	11.900000	0.080000	165.362640
75%	-9.262278	53.228333	12.500000	303.912080	7.080000	249.000000	190.000000	14.730000	0.137000	268.219800
max	-9.262278	54.275300	25.000000	359.648350	21.970000	1819.000000	1407.000000	22.360000	1.363000	359.912080

At a first look, we can see that maybe there are outliers, let's see it more visually.



Now we can see more clearly that there are some outliers, but they look like more a natural event than an error on the measuring. Therefore, we will not perform any changes.

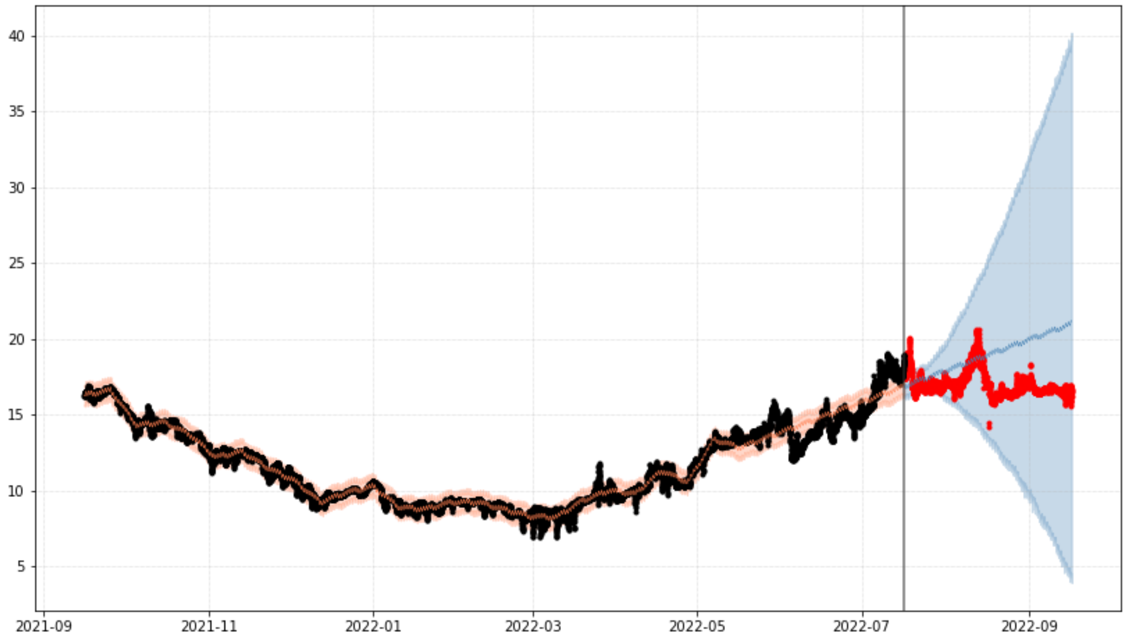
Model

I chose to the Prophet library for creating my model and analyzing the data trends.

Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well.

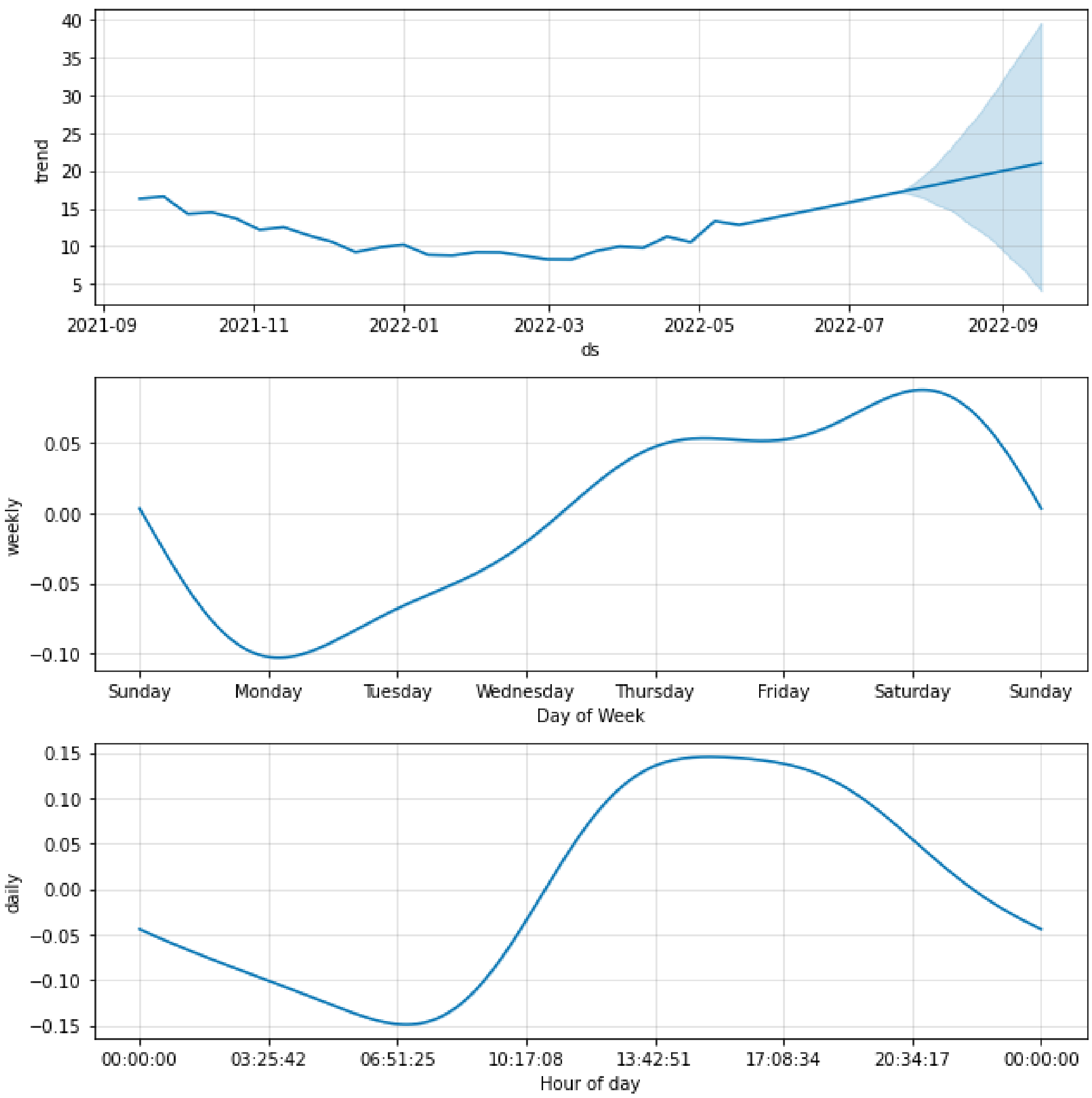
Since the data is not symmetrically spaced, I grouped it into a 30 minutes interval and get the mean of the period

But why 30 min? We see that our max interval between dates were 30 min, if we get less than that, there would be some nan values.



In black we have the train data, which is the real Sea Temperature through time, in orange we have the predictions.

In red we have the test data, which is 2 months of data that were not used for training. In blue we have the predictions of our model.



In these charts we have the characteristics of our model and data.

We can see that the global trending is that the temperature will raise.

The temperature tends to raise throughout the week, reaching its highest on Saturday and the lowest on Monday.

Throughout the day, it reaches his highest values between 13:40 and 17:00, during the afternoon, period where there is more sun energy power.

Evaluate

To evaluate our model, we calculate some metrics regarding to the train and test sets.

	rmse_train	mae_train	mape_train	r2_train	rmse_test	mae_test	mape_test	r2_test
0	0.50866	0.342025	0.027945	0.963163	2.526067	2.097714	0.125514	-8.943772

The model seems overfit a lot, and it doesn't provide good predictions. On the train data it provides very good results, with low RMSE and a high R2 Score, but in the test data the metrics get a lot worse.

We can see on the chart that it captures the tendency to grow, but the model ignores that there is a strong seasonality due to the seasons of the year and the temperature can't continuously grow like that.

Another way to improve the model it was passing the location, enhancing the predict.

The model needs to be optimized to capture these seasonalities and predict with more accuracy. Since we only collected data for a year, I am sure that if we had data on more years, the model would capture these patterns in a better way.

4.2 Group the oceans using a machine learning model suited to this task.

In order to group our data, we grouped each station by it's mean, to get the overall of each feature.

	station_id	latitude	longitude	Water_Level_LAT	Water_Level_OD_Malin	QC_Flag
0	Aranmore Island - Leabgarrow	54.990500	-8.49550	2.237245	0.034245	0.930047
1	Ballycotton Harbour	51.827800	-8.00070	2.384431	-0.104569	0.871555
2	Ballyglass Harbour	54.253600	-9.89280	2.024153	-0.079847	0.935218
3	Castletownbere Port	51.649600	-9.90340	1.882076	-0.207924	0.920984
4	Dingle Harbour	52.139240	-10.27732	2.489094	-0.065906	0.931706
5	Dublin Port	53.345700	-6.22170	2.519070	0.014070	0.000000
6	Galway Port	53.269000	-9.04800	2.999361	0.032361	0.935729
7	Howth Water Level 1	53.391335	-6.06809	NaN	-0.080356	0.933877
8	Killybegs Port	54.636400	-8.39490	2.332173	0.039173	0.917704
9	Kinvara - Unreferenced	53.140520	-8.93758	NaN	-2.365941	0.831120
10	Roonagh Pier	53.762350	-9.90442	2.425708	0.470708	0.000000
11	Skerries Harbour	53.585000	-6.10810	2.768237	-0.090763	0.000000
12	Sligo	54.309900	-8.58200	2.265840	0.020840	0.892706
13	Union Hall Harbor	51.558964	-9.13349	NaN	-0.204473	0.935398
14	Wexford Harbour	52.338500	-6.45890	0.861423	-0.094577	0.935892

<u>Model</u>

We will be using the model that is considered one of the simplest models amongst clustering. Despite its simplicity, the **K-means** is vastly used for clustering in many data science applications, it is especially useful if you need to quickly discover insights from **unlabeled data**.

We instantiated our model to divided the data into 4 clusters.

	Water_Level_OD_Malin								QC_Flag							
	count	mean	std	min	25%	50%	75%	max	count	mean	std	min	25%	50%	75%	max
Cluster Labels																
0	11.0	-0.064639	0.089624	-0.207924	-0.099573	-0.079847	0.026600	0.039173	11.0	0.921892	0.021114	0.871555	0.919344	0.931706	0.935308	0.935892
1	1.0	-2.365941	NaN	-2.365941	-2.365941	-2.365941	-2.365941	-2.365941	1.0	0.831120	NaN	0.831120	0.831120	0.831120	0.831120	0.831120
2	2.0	-0.038346	0.074128	-0.090763	-0.064554	-0.038346	-0.012138	0.014070	2.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
3	1.0	0.470708	NaN	0.470708	0.470708	0.470708	0.470708	0.470708	1.0	0.000000	NaN	0.000000	0.000000	0.000000	0.000000	0.000000

Analyzing the statistics:

- Cluster nº 0 have the majority of the stations in it, this means that the stations were too similar or we don't have enough data about it to differentiate them
- We see that the cluster nº 3 have the highest water level mean between them, and the cluster nº 1 have the lowest.

Here we can have an overview of how each cluster is on the coast.

In Red we have cluster nº 0.

In Green we have cluster nº 1.

In Yellow we have cluster nº 2.

In Grey we have cluster nº 3.

Looking at the map, the clusters didn't seem to be influenced by the localization of the station.

To improve the analysis, we would need more features about each station, so the model could better capture the patterns.