Exploratory Data Analysis

Dataset Name - Algerian Forest Fire

0) Dataset download link

Dataset Link - [https://archive.ics.uci.edu/ml/datasets/Algerian+Forest+Fires+Dataset++ (https://archive.ics.uci.edu/ml/datasets/Algerian+Forest+Fires+Dataset++)](https://archive.ics.uci.edu/ml/datasets/Algerian+Forest+Fires+Dataset++)

1) Problem statement.

```
i)  The period from June 2012 to September 2012.
ii) The dataset includes 11 attribues and 1 output attribue.
iii)Total 12 attributes are there.
iv) Target variable is fire.
```

2) Data Collection

Importing Required Libraries

In [1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import plotly.express as px
import matplotlib.pyplot as plt
```

Reading dataset

In [2]:
```
df = pd.read_csv("forest_fire.csv")
df.drop(index=[122,123,124], inplace=True)
df.reset_index(drop=True, inplace=True)
df.head()
```

Out[2]:

	day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Classes
0	1	6	2012	29	57	18	0	65.7	3.4	7.6	1.3	3.4	0.5	not fire
1	2	6	2012	29	61	13	1.3	64.4	4.1	7.6	1	3.9	0.4	not fire
2	3	6	2012	26	82	22	13.1	47.1	2.5	7.1	0.3	2.7	0.1	not fire
3	4	6	2012	25	89	13	2.5	28.6	1.3	6.9	0	1.7	0	not fire
4	5	6	2012	27	77	16	0	64.8	3	14.2	1.2	3.9	0.5	not fire

Shape of dataset

In [3]:
```
df.shape
```

Out[3]: (244, 14)

```
i) We have 244 rows and 14 columns
```

Statistics Summary of the dataset

In [4]:
```
df.describe(include = 'all')
```

Out[4]:

	day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Classes
count	244	244	244	244	244	244	244	244	244	244	244	244	244	243
unique	31	4	1	19	62	18	39	173	166	198	106	174	126	8
top	1	7	2012	35	64	14	0	88.9	7.9	8	1.1	3	0.4	fire
freq	8	62	244	29	10	43	133	8	5	5	8	5	12	131

In [5]:
```
# Check Null and Dtypes
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 244 entries, 0 to 243
Data columns (total 14 columns):
 #   Column       Non-Null Count  Dtype 
---  ------       --------------  ----- 
 0   day          244 non-null    object
 1   month        244 non-null    object
 2   year         244 non-null    object
 3   Temperature  244 non-null    object
 4    RH          244 non-null    object
 5    Ws          244 non-null    object
 6   Rain         244 non-null    object
 7   FFMC         244 non-null    object
 8   DMC          244 non-null    object
 9   DC           244 non-null    object
 10  ISI          244 non-null    object
 11  BUI          244 non-null    object
 12  FWI          244 non-null    object
 13  Classes      243 non-null    object
dtypes: object(14)
memory usage: 26.8+ KB
```

As the Dtype is object we not able to see all the details in describe so let's first convert it into numeric data

In [24]:
```
df.replace('14.6 9', '14.69',inplace=True)
df.drop(index=165,inplace=True)
df.reset_index(drop=True, inplace=True)
df.head(167)
```

Out[24]:

	day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Under_Fire	Region	Classes
0	1	6	2012	29	57	18	0	65.7	3.4	7.6	1.3	3.4	0.5	not fire	0	notfire
1	2	6	2012	29	61	13	1.3	64.4	4.1	7.6	1	3.9	0.4	not fire	0	notfire
2	3	6	2012	26	82	22	13.1	47.1	2.5	7.1	0.3	2.7	0.1	not fire	0	notfire
3	4	6	2012	25	89	13	2.5	28.6	1.3	6.9	0	1.7	0	not fire	0	notfire
4	5	6	2012	27	77	16	0	64.8	3	14.2	1.2	3.9	0.5	not fire	0	notfire
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
162	11	7	2012	34	56	15	2.9	74.8	7.1	9.5	1.6	6.8	0.8	not fire	1	notfire
163	12	7	2012	36	44	13	0	90.1	12.6	19.4	8.3	12.5	9.6	fire	1	fire
164	13	7	2012	39	45	13	0.6	85.2	11.3	10.4	4.2	10.9	4.7	fire	1	fire
165	15	7	2012	34	45	17	0	90.5	18	24.1	10.9	17.7	14.1	fire	1	fire
166	16	7	2012	31	83	17	0	84.5	19.4	33.1	4.7	19.2	7.3	fire	1	fire

167 rows × 16 columns

In [34]:
```
df.drop(columns=['Under_Fire'],inplace=True )
```

In [35]:
```
df.set_axis(['day','month','year','Temperature','RH','Ws','Rain','FFMC','DMC','DC','ISI','BUI','FWI','Region','Classes']
```

In [37]:
```
df
```

Out[37]:

	day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Region	Classes
0	1	6	2012	29	57	18	0	65.7	3.4	7.6	1.3	3.4	0.5	0	notfire
1	2	6	2012	29	61	13	1.3	64.4	4.1	7.6	1	3.9	0.4	0	notfire
2	3	6	2012	26	82	22	13.1	47.1	2.5	7.1	0.3	2.7	0.1	0	notfire
3	4	6	2012	25	89	13	2.5	28.6	1.3	6.9	0	1.7	0	0	notfire
4	5	6	2012	27	77	16	0	64.8	3	14.2	1.2	3.9	0.5	0	notfire
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
238	26	9	2012	30	65	14	0	85.4	16	44.5	4.5	16.9	6.5	1	fire
239	27	9	2012	28	87	15	4.4	41.1	6.5	8	0.1	6.2	0	1	notfire
240	28	9	2012	27	87	29	0.5	45.9	3.5	7.9	0.4	3.4	0.2	1	notfire
241	29	9	2012	24	54	18	0.1	79.7	4.3	15.2	1.7	5.1	0.7	1	notfire
242	30	9	2012	24	64	15	0.2	67.3	3.8	16.5	1.2	4.8	0.5	1	notfire

243 rows × 15 columns

In [41]:
```
df.columns
```

Out[41]:
```
Index(['day', 'month', 'year', 'Temperature', 'RH', 'Ws', 'Rain', 'FFMC',
       'DMC', 'DC', 'ISI', 'BUI', 'FWI', 'Region', 'Classes'],
      dtype='object')
```

In [50]:
```
for feature in ['Classes']:
    df[feature] = df[feature].str.replace(' ', '')
```

In [51]:
```
df['Classes'].unique()
```

Out[51]:
```
array(['notfire', 'fire'], dtype=object)
```

In [56]:
```
### changing datatypes of features to numerical for numerical features as all are in object

datatype_convert={'day':'int64','month':'int64','year':'int64','Temperature':'int64','RH':'int64', 'Ws':'int64','Rain':'
                  'FFMC':'float64', 'DMC':'float64', 'DC':'float64', 'ISI':'float64', 'BUI':'float64', 'FWI':'float64',
                 'Region':'float64'}

df=df.astype(datatype_convert)
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 243 entries, 0 to 242
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   day          243 non-null    int64  
 1   month        243 non-null    int64  
 2   year         243 non-null    int64  
 3   Temperature  243 non-null    int64  
 4   RH           243 non-null    int64  
 5   Ws           243 non-null    int64  
 6   Rain         243 non-null    float64
 7   FFMC         243 non-null    float64
 8   DMC          243 non-null    float64
 9   DC           243 non-null    float64
 10  ISI          243 non-null    float64
 11  BUI          243 non-null    float64
 12  FWI          243 non-null    float64
 13  Region       243 non-null    float64
 14  Classes      243 non-null    object 
dtypes: float64(8), int64(6), object(1)
memory usage: 28.6+ KB
```

In [57]:
```
df
```

Out[57]:

	day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Region	Classes
0	1	6	2012	29	57	18	0.0	65.7	3.4	7.6	1.3	3.4	0.5	0.0	notfire
1	2	6	2012	29	61	13	1.3	64.4	4.1	7.6	1.0	3.9	0.4	0.0	notfire
2	3	6	2012	26	82	22	13.1	47.1	2.5	7.1	0.3	2.7	0.1	0.0	notfire
3	4	6	2012	25	89	13	2.5	28.6	1.3	6.9	0.0	1.7	0.0	0.0	notfire
4	5	6	2012	27	77	16	0.0	64.8	3.0	14.2	1.2	3.9	0.5	0.0	notfire
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
238	26	9	2012	30	65	14	0.0	85.4	16.0	44.5	4.5	16.9	6.5	1.0	fire
239	27	9	2012	28	87	15	4.4	41.1	6.5	8.0	0.1	6.2	0.0	1.0	notfire
240	28	9	2012	27	87	29	0.5	45.9	3.5	7.9	0.4	3.4	0.2	1.0	notfire
241	29	9	2012	24	54	18	0.1	79.7	4.3	15.2	1.7	5.1	0.7	1.0	notfire
242	30	9	2012	24	64	15	0.2	67.3	3.8	16.5	1.2	4.8	0.5	1.0	notfire

243 rows × 15 columns

In [58]:
```
df.describe()
```

Out[58]:

	day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BU
count	243.000000	243.000000	243.0	243.000000	243.000000	243.000000	243.000000	243.000000	243.000000	243.000000	243.000000	243.00000
mean	15.761317	7.502058	2012.0	32.152263	62.041152	15.493827	0.762963	77.842387	14.680658	49.430864	4.742387	16.69053
std	8.842552	1.114793	0.0	3.628039	14.828160	2.811385	2.003207	14.349641	12.393040	47.665606	4.154234	14.22842
min	1.000000	6.000000	2012.0	22.000000	21.000000	6.000000	0.000000	28.600000	0.700000	6.900000	0.000000	1.10000
25%	8.000000	7.000000	2012.0	30.000000	52.500000	14.000000	0.000000	71.850000	5.800000	12.350000	1.400000	6.00000
50%	16.000000	8.000000	2012.0	32.000000	63.000000	15.000000	0.000000	83.300000	11.300000	33.100000	3.500000	12.40000
75%	23.000000	8.000000	2012.0	35.000000	73.500000	17.000000	0.500000	88.300000	20.800000	69.100000	7.250000	22.65000
max	31.000000	9.000000	2012.0	42.000000	90.000000	29.000000	16.800000	96.000000	65.900000	220.400000	19.000000	68.00000

3) Exploring Data

In [59]:
```
# define numerical & categorical columns
numeric_features = [feature for feature in df.columns if df[feature].dtype != 'O']
categorical_features = [feature for feature in df.columns if df[feature].dtype == 'O']

# print columns
print('We have {} numerical features : {}'.format(len(numeric_features), numeric_features))
print('\nWe have {} categorical features : {}'.format(len(categorical_features), categorical_features))
```

```
We have 14 numerical features : ['day', 'month', 'year', 'Temperature', 'RH', 'Ws', 'Rain', 'FFMC', 'DMC', 'DC', 'ISI',
'BUI', 'FWI', 'Region']

We have 1 categorical features : ['Classes']
```

Attribute information

```
1. Date : (DD/MM/YYYY) Day, month ('june' to 'september'), year (2012)
Weather data observations
2. Temp : temperature noon (temperature max) in Celsius degrees: 22 to 42
3. RH : Relative Humidity in %: 21 to 90
4. Ws :Wind speed in km/h: 6 to 29
5. Rain: total day in mm: 0 to 16.8
FWI Components
6. Fine Fuel Moisture Code (FFMC) index from the FWI system: 28.6 to 92.5
7. Duff Moisture Code (DMC) index from the FWI system: 1.1 to 65.9
8. Drought Code (DC) index from the FWI system: 7 to 220.4
9. Initial Spread Index (ISI) index from the FWI system: 0 to 18.5
10. Buildup Index (BUI) index from the FWI system: 1.1 to 68
11. Fire Weather Index (FWI) Index: 0 to 31.1
12. Classes: two classes, namely Not Fire (0) and Fire(1)
```

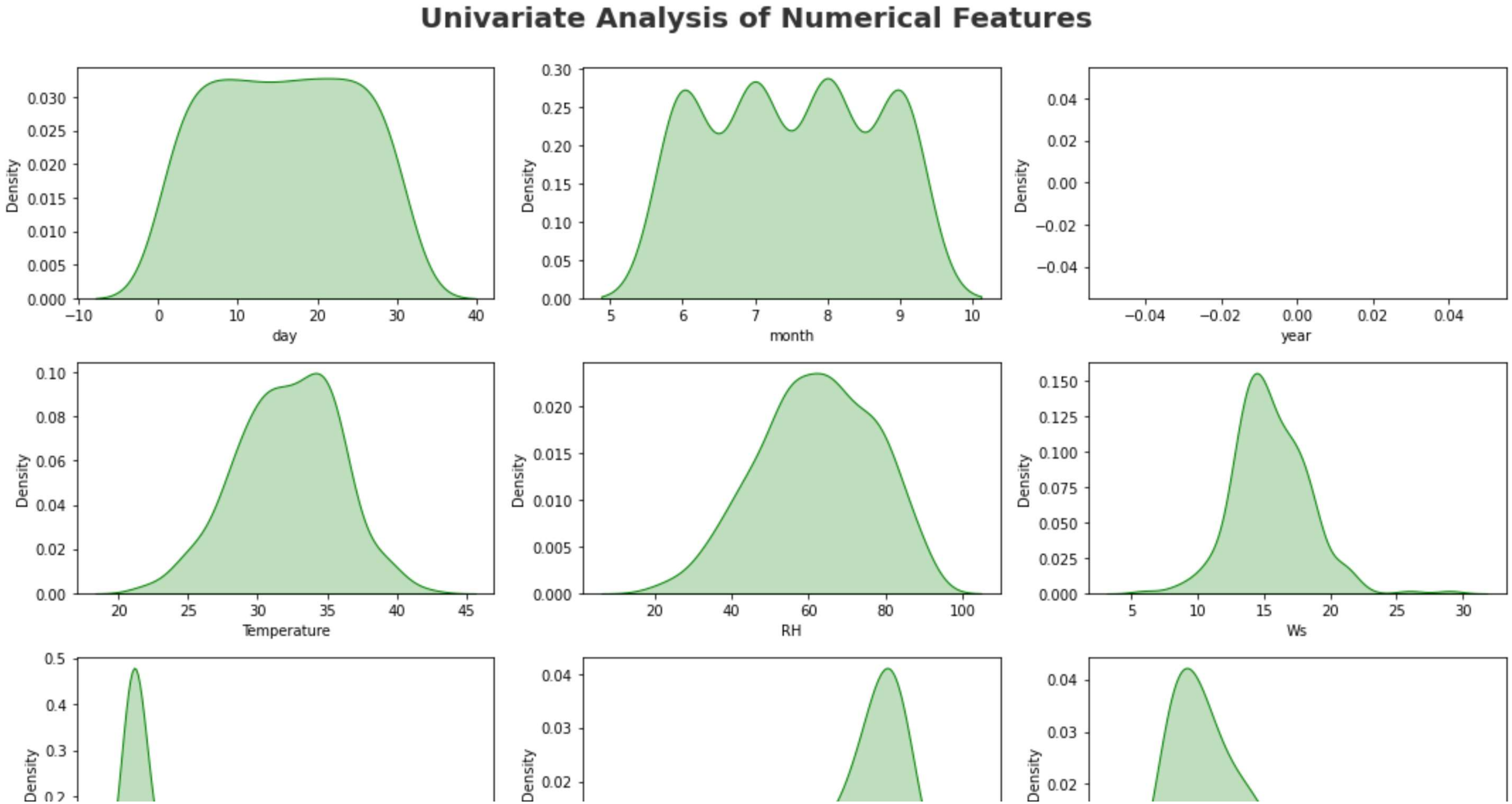
Univarite Analysis

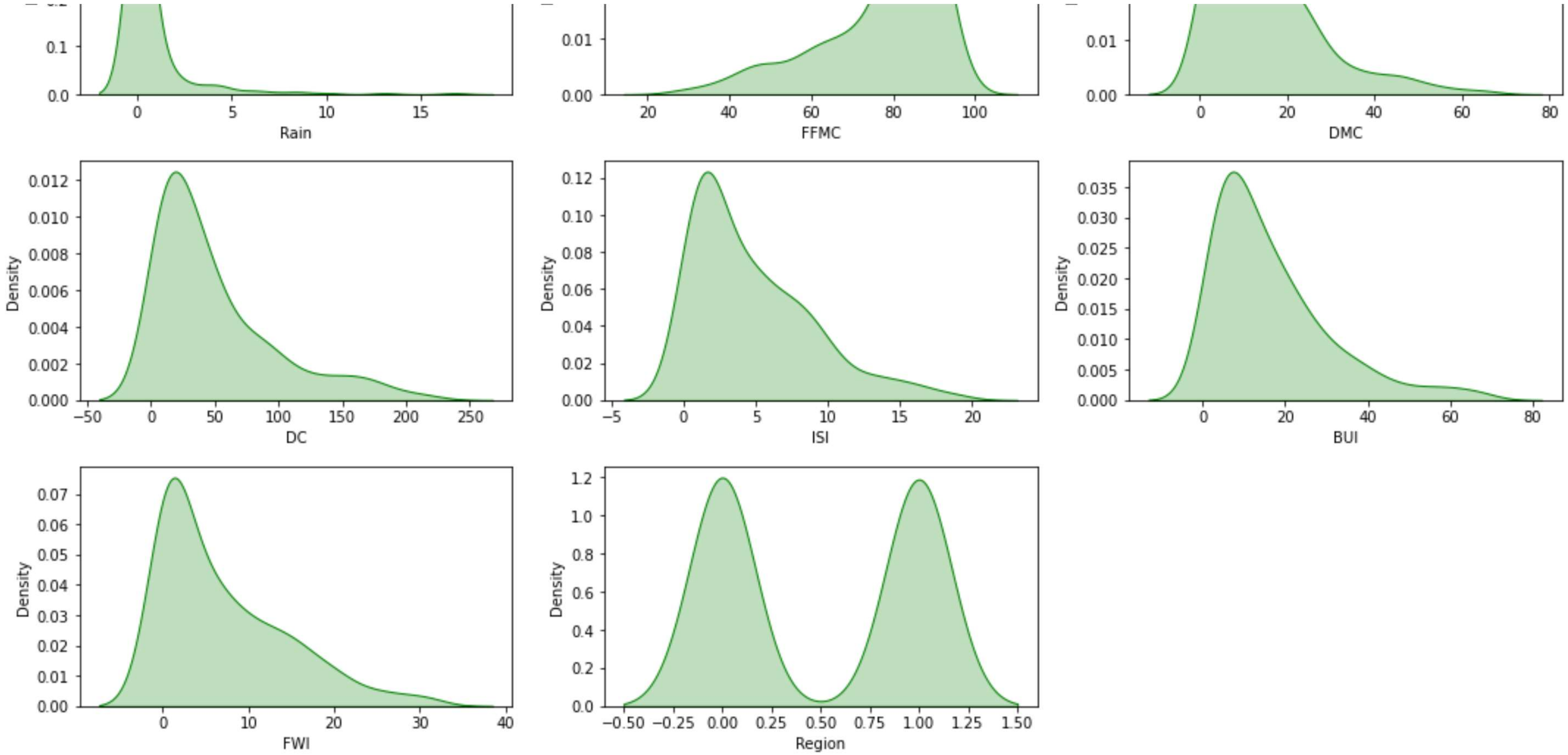
```
In univariate analysis we are doing analysis on single column/feature, basically we are trying to see the distribution
of datapoints.
```

In [60]:
```
plt.figure(figsize=(15, 15))
plt.suptitle('Univariate Analysis of Numerical Features', fontsize=20, fontweight='bold', alpha=0.8, y=1.)

for i in range(0, len(numeric_features)):
    plt.subplot(5, 3, i+1)
    sns.kdeplot(x=df[numeric_features[i]],shade=True, color='g')
    plt.xlabel(numeric_features[i])
    plt.tight_layout()
```

```
c:\users\dell\appdata\local\programs\python\python38\lib\site-packages\seaborn\distributions.py:316: UserWarning: Datas
et has 0 variance; skipping density estimate. Pass `warn_singular=False` to disable this warning.
  warnings.warn(msg, UserWarning)
```





Observation from above graphs

```
i) FWI, BUI, ISI, DC, DMC, Rain are right skewed.
ii) FFMC is left skewed.
iii) Temprature, WS, RH are almost normal distributed.
```

Multivariate Analysis

```
Multivariate Analysis means we are comparing two or more column/feature.
```

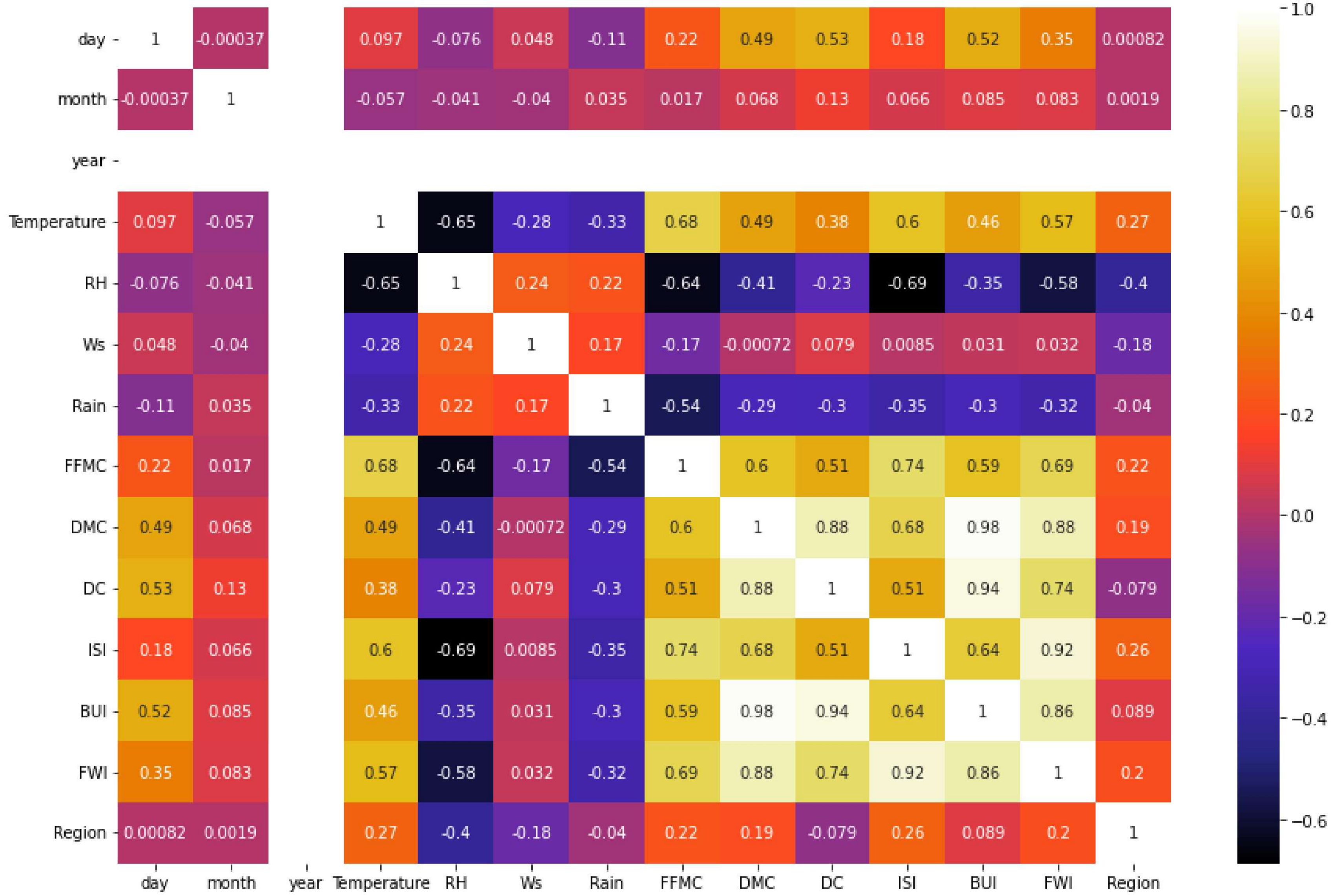
Checking Multicolinarity

In [61]:
```
df[(list(df.columns)[1:])].corr()
```

Out[61]:

	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Regic
month	1.000000	NaN	-0.056781	-0.041252	-0.039880	0.034822	0.017030	0.067943	0.126511	0.065608	0.085073	0.082639	0.00185
year	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na
Temperature	-0.056781	NaN	1.000000	-0.651400	-0.284510	-0.326492	0.676568	0.485687	0.376284	0.603871	0.459789	0.566670	0.26955
RH	-0.041252	NaN	-0.651400	1.000000	0.244048	0.222356	-0.644873	-0.408519	-0.226941	-0.686667	-0.353841	-0.580957	-0.40268
Ws	-0.039880	NaN	-0.284510	0.244048	1.000000	0.171506	-0.166548	-0.000721	0.079135	0.008532	0.031438	0.032368	-0.18116
Rain	0.034822	NaN	-0.326492	0.222356	0.171506	1.000000	-0.543906	-0.288773	-0.298023	-0.347484	-0.299852	-0.324422	-0.04001
FFMC	0.017030	NaN	0.676568	-0.644873	-0.166548	-0.543906	1.000000	0.603608	0.507397	0.740007	0.592011	0.691132	0.22224
DMC	0.067943	NaN	0.485687	-0.408519	-0.000721	-0.288773	0.603608	1.000000	0.875925	0.680454	0.982248	0.875864	0.19208
DC	0.126511	NaN	0.376284	-0.226941	0.079135	-0.298023	0.507397	0.875925	1.000000	0.508643	0.941988	0.739521	-0.07873
ISI	0.065608	NaN	0.603871	-0.686667	0.008532	-0.347484	0.740007	0.680454	0.508643	1.000000	0.644093	0.922895	0.26319
BUI	0.085073	NaN	0.459789	-0.353841	0.031438	-0.299852	0.592011	0.982248	0.941988	0.644093	1.000000	0.857973	0.08940
FWI	0.082639	NaN	0.566670	-0.580957	0.032368	-0.324422	0.691132	0.875864	0.739521	0.922895	0.857973	1.000000	0.19710
Region	0.001857	NaN	0.269555	-0.402682	-0.181160	-0.040013	0.222241	0.192089	-0.078734	0.263197	0.089408	0.197102	1.00000

In [62]:
```
plt.figure(figsize = (15,10))
sns.heatmap(df.corr(), cmap="CMRmap", annot=True)
plt.show()
```



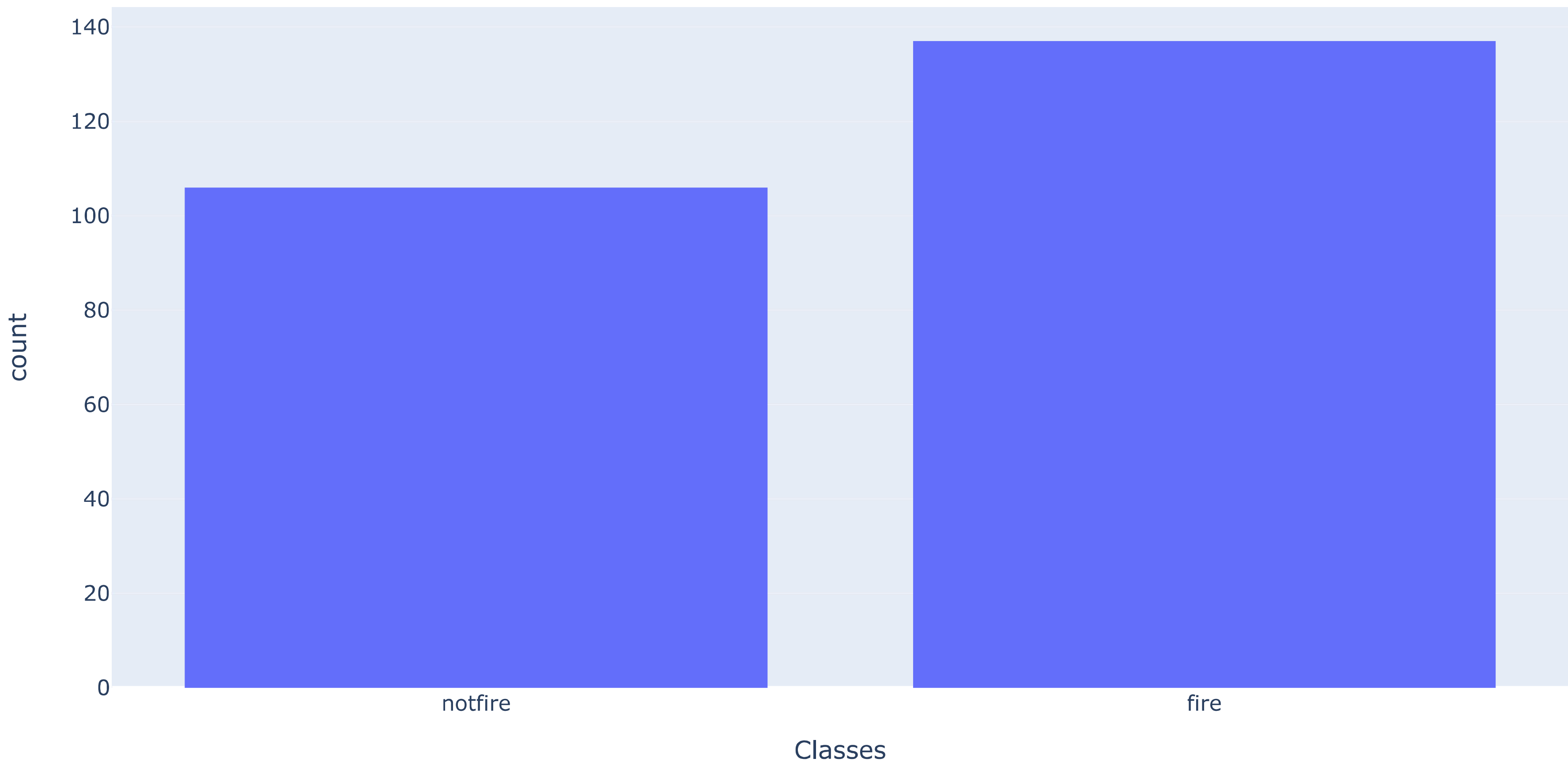
Observation from above heatmap

```
Classes varibale i.e. our target variabale/feature has negative correlation with the Rain,Ws,RH
Classes variable has high correlation with the FFMC,ISI,FWI and somewhat with DMC and DC
```

4) Visualization

4.1 Visualize the Target Feature

In [71]:
```
px.histogram(df, x='Classes')
```



Observation from above histogram

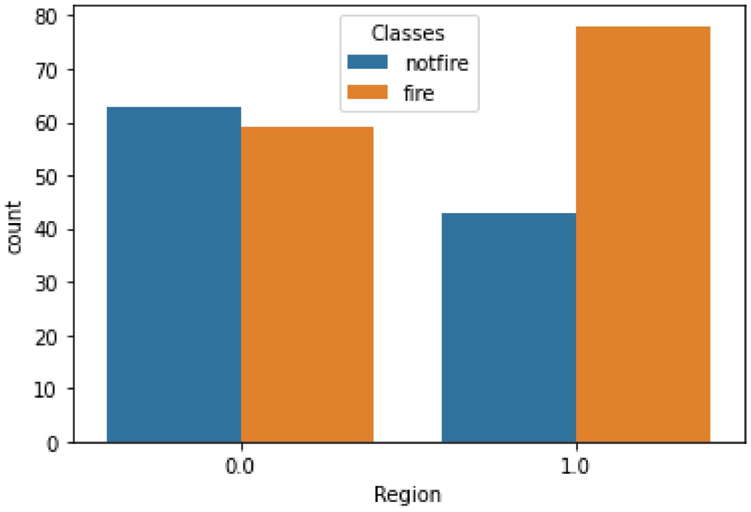
```
 0 - No Fire
 1 - Fire
As per the graph No fire count is 106
while the fire count is 137
```

```
this means that fire is more likely to happen than no fire in forest
```

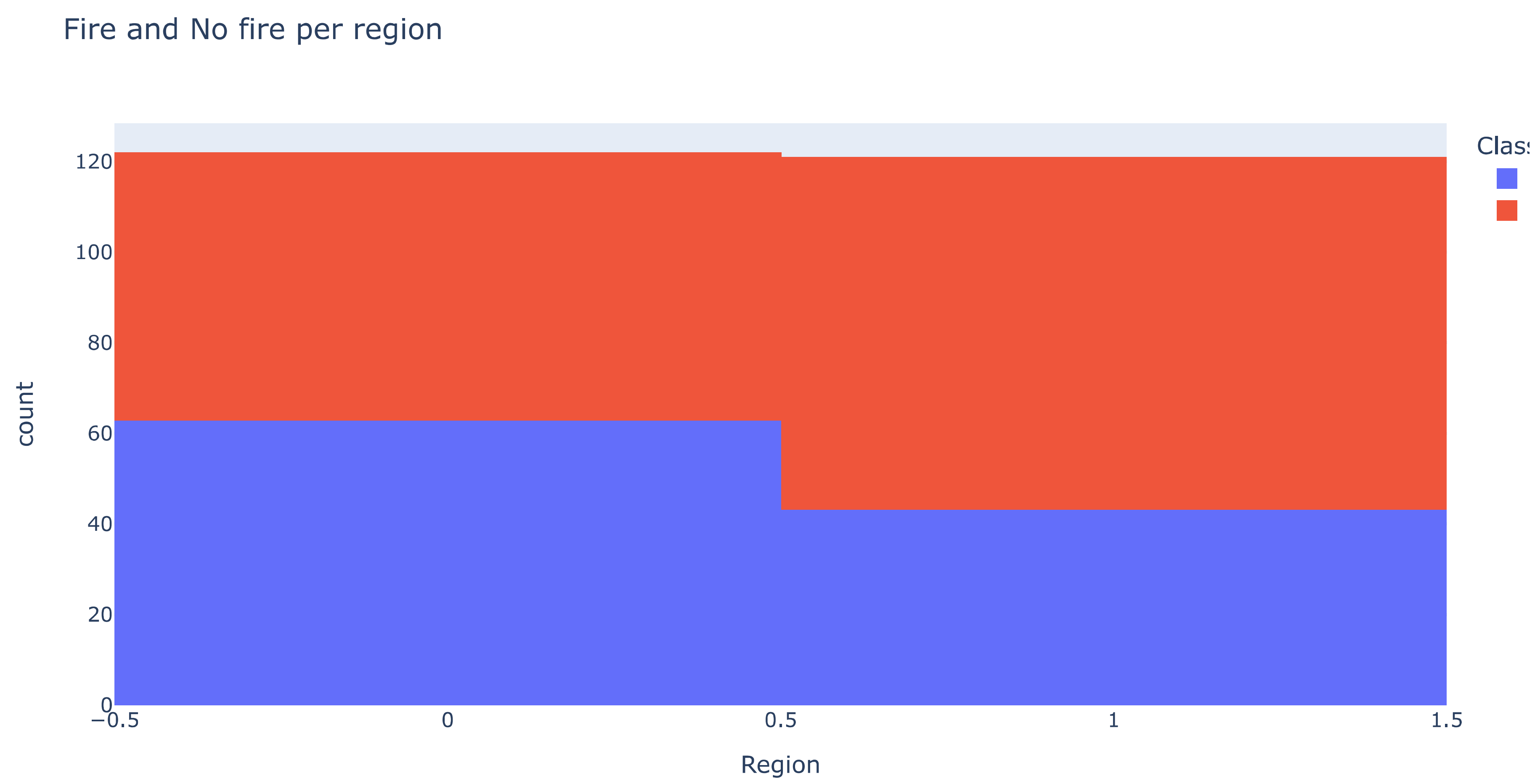
4.2) Let's see fire and no fire per region

In [79]:
```
sns.countplot(data = df, x = 'Region', hue = 'Classes')
```

Out[79]:
```
<AxesSubplot:xlabel='Region', ylabel='count'>
```



In [84]:
```
px.histogram(df, x='Region',color='Classes', title="Fire and No fire per region")
```



Observation from above graph

```
Region 0 -  Bejaia region
Region 1 -  Sidi Bel-abbes region
i) Region 0 :
        Fire Count - 59
```

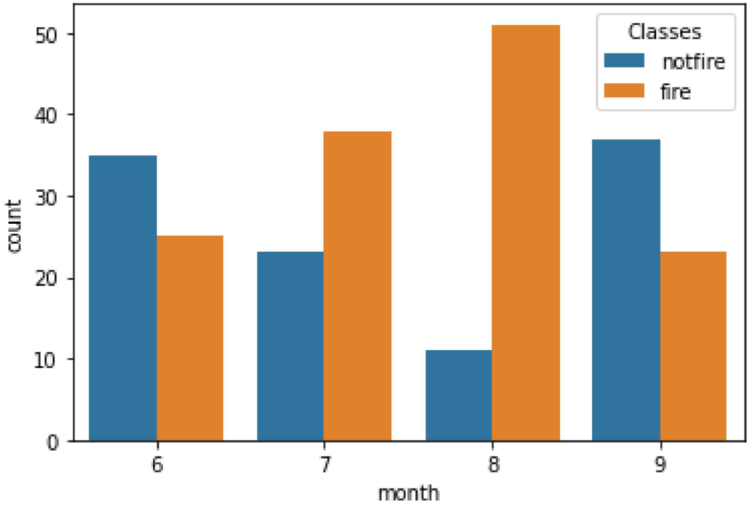
```
        Not Fire Count - 63
ii) Region 1 :
        Fire Count - 78
        Not Fire Count - 43

Region 0 has more fire count than region 1
```

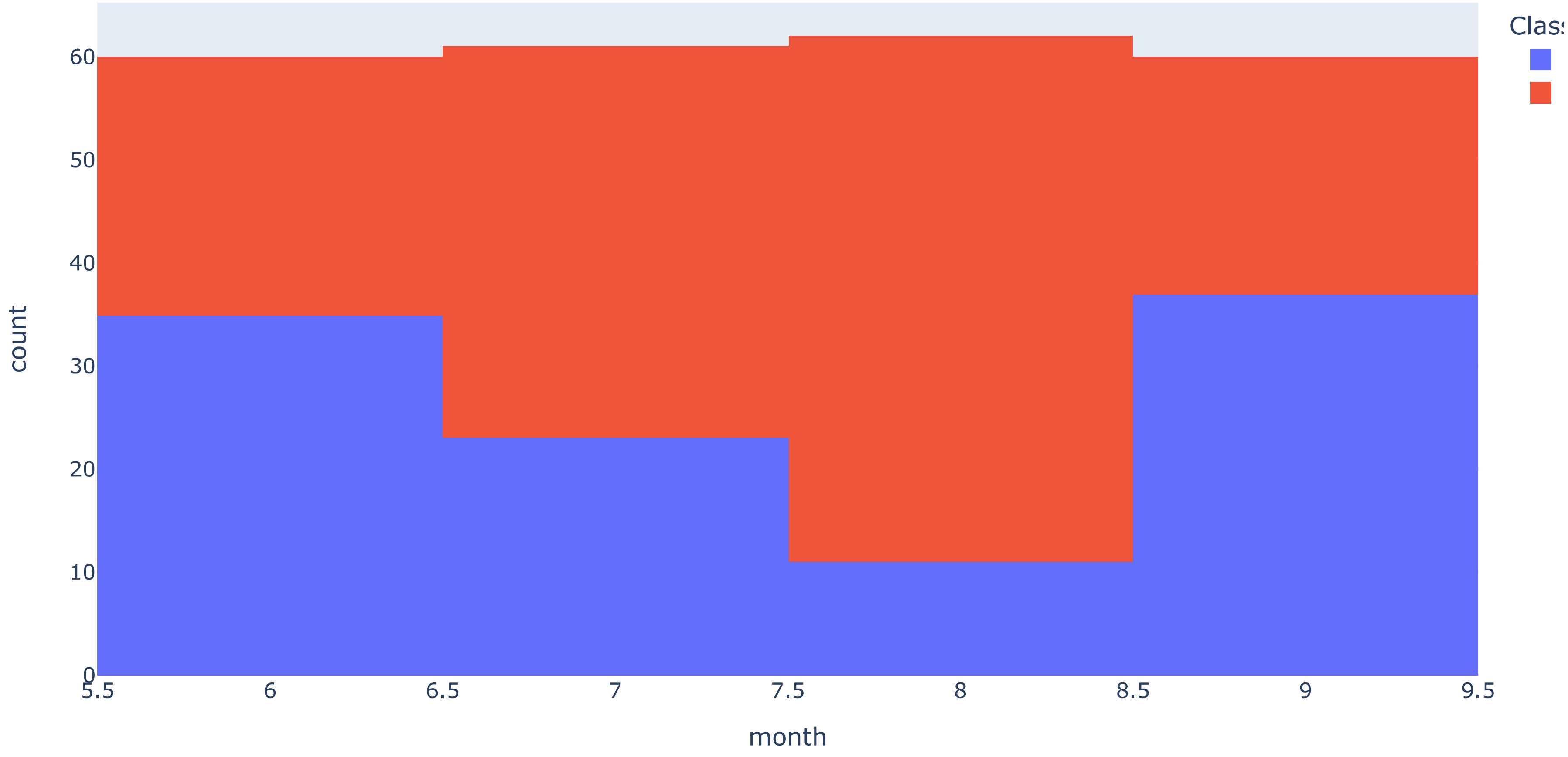
4.3)Fire Per month

In [89]:
```
sns.countplot(data=df, x='month', hue='Classes')
```

Out[89]:
```
<AxesSubplot:xlabel='month', ylabel='count'>
```



In [94]:
```
px.histogram(df, x="month", color="Classes")
```

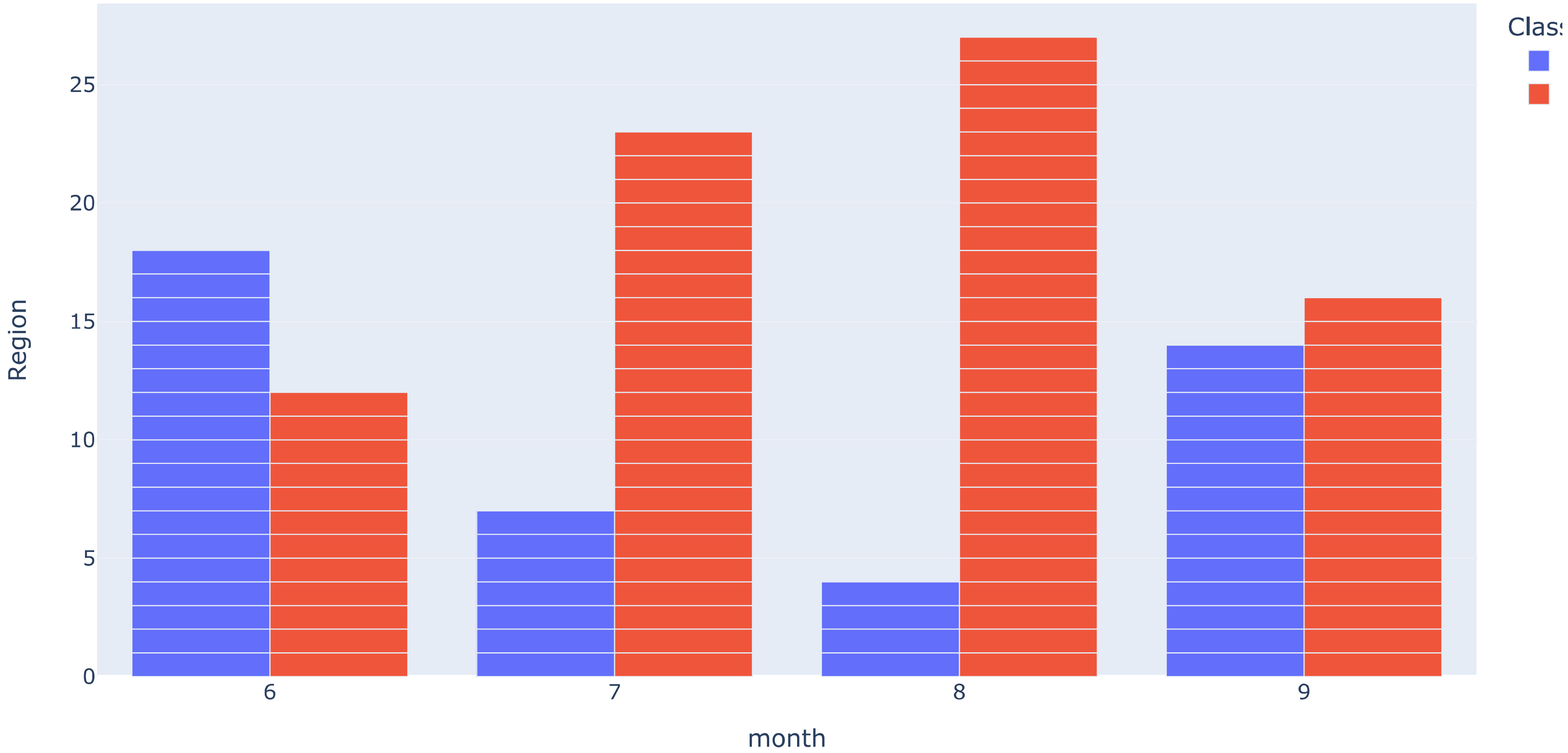


Observation from above histogram

```
8th Month has highest number of fire count 51 followed by 7th Month 38 and 6th month 25
9th Month has highest number of not fire count 37
```

4.4) Fire per region per Month

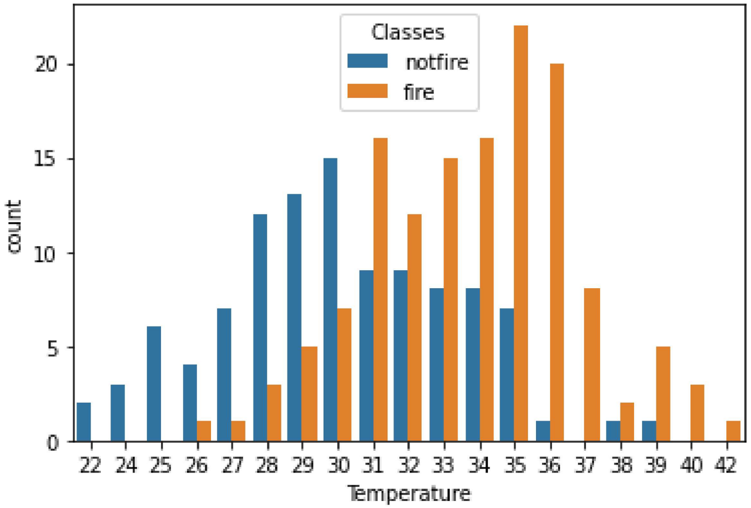
In [106]:
```
px.bar(df, x="month",y='Region',color="Classes", barmode='group' )
```



4.5) Relation of Temprature with fire

In [123]:
```
sns.countplot(data=df, x='Temperature', hue='Classes')
```

Out[123]: <AxesSubplot:xlabel='Temperature', ylabel='count'>



Observartion from above graph

```
as the temprature increases the chances of fire are also increases.
when the temprature increases above 30 degree celcius the no of chances of fire increases very high
```

5) Conclusion

In [ ]:
```
1) FWI, BUI, ISI, DC, DMC, Rain are right skewed.
2) FFMC is left skewed.
3) Temprature, WS, RH are almost normal distributed.
4)Classes varibale i.e. our target variabale/feature has negative correlation with the Rain,Ws,RH
5)Classes variable has high correlation with the FFMC,ISI,FWI and somewhat with DMC and DC
6)No fire count is 106
7)the fire count is 137
8)this means that fire is more likely to happen than no fire
Region 0 -  Bejaia region
Region 1 -  Sidi Bel-abbes region
i) Region 0 :
        Fire Count - 59
        Not Fire Count - 63
ii) Region 1 :
        Fire Count - 78
        Not Fire Count - 43

Region 0 has more fire count than region 1
9)8th Month has highest number of fire count 51 followed by 7th Month 38 and 6th month 25
10)9th Month has highest number of not fire count 37
11)as the temprature increases the chances of fire are also increases.
12)when the temprature increases above 30 degree celcius the no of chances of fire increases very high
```