ĐẠI HỌC BÁCH KHOA HÀ NỘI

TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

*****

BÁO CÁO

Nhập môn khoa học dữ liệu

ĐỀ TÀI:

Xây dựng mô hình dự đoán thời tiết ở Hà Nội

GVHD: TS. Trần Việt Trung

Sinh viên thực hiện:

Họ tên	**MSSV**
1. Nguyễn Trần Chung	20204520
2. Trần Nguyễn Anh Tuấn	20200565
3. Nguyễn Hoàng Hải	20204648
4. Nguyễn Gia Khánh	20204661

Hà Nội, tháng 12 năm 2023

CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

Tiếp cận và ứng dụng khoa học dữ liệu đã trở thành một xu hướng quan trọng trong nhiều lĩnh vực khác nhau, và lĩnh vực dự đoán thời tiết không ngoại lệ. Dự đoán chính xác các điều kiện thời tiết không chỉ là một nhiệm vụ quan trọng cho các cơ quan khí tượng mà còn có thể ảnh hưởng đáng kể đến cuộc sống hàng ngày của chúng ta. Trong báo cáo này, nhóm chúng em tập trung vào dự đoán thời tiết tại Hà Nội, thủ đô của Việt Nam, sử dụng phương pháp và kỹ thuật của khoa học dữ liệu.

Hà Nội, với khí hậu nhiệt đới ẩm và bị ảnh hưởng bởi sự biến đổi khí hậu toàn cầu, thường xuyên trải qua các biến động thời tiết đáng kể. Khả năng dự đoán chính xác các thay đổi này có thể cung cấp thông tin quan trọng cho các lĩnh vực như nông nghiệp, giao thông, du lịch và quản lý tài nguyên tự nhiên.

Trong quá trình nghiên cứu này, nhóm chúng em tập trung vào việc khám phá và phân tích dữ liệu thời tiết lịch sử để xây dựng mô hình dự đoán thời tiết cho Hà Nội. Nhóm sử dụng một tập dữ liệu lớn về các biến số thời tiết như nhiệt độ, độ ẩm, áp suất không khí và tốc độ gió được thu thập từ các trạm quan trắc khí tượng trong suốt một khoảng thời gian dài. Bằng cách áp dụng các kỹ thuật phân tích dữ liệu và mô hình hóa, nhóm chúng em hy vọng xây dựng một mô hình dự đoán thời tiết chính xác và tin cậy cho Hà Nội.

CHƯƠNG 2: ĐẶT VẤN ĐỀ VÀ YÊU CẦU

2.1 Đặt vấn đề

Hiểu và dự đoán điều kiện thời tiết có tầm quan trọng lớn trong nhiều lĩnh vực khác nhau, chẳng hạn như nông nghiệp, giao thông, lên sự kiện,... Dự đoán chính xác có thể hỗ trợ đưa ra quyết định sáng suốt, giảm thiểu rủi ro và tối ưu hóa việc phân bổ nguồn lực. Trước đây ở Việt Nam, người ta dự đoán thời tiết theo đặc trưng các mùa, quan sát cảnh vật xung quanh; song ngày nay, bằng những công cụ cảm biến tiên tiến, chúng ta có thêm nhiều nguồn dữ liệu để góp vào thành những đặc trưng thể hiện cho thời tiết hiện tại và tương lai gần.

Trong bài tập lớn môn học "Nhập môn Khoa học dữ liệu", nhóm sẽ tiến hành thu thập dữ liệu thời tiết từ nhiều nơi, theo nhiều năm; phân tích những đặc trưng lẫn tác động của chúng lẫn nhau lên thời tiết ở thời gian cụ thể; và sử dụng chúng trong các mô hình học máy-học sâu trong dự đoán thời tiết.

2.2 Yêu cầu

Tìm một nguồn dữ liệu đáng tin cậy về thời tiết, có thông tin chi tiết về hướng gió, lượng bức xạ, lượng mưa, … để đưa ra các phán đoán chính xác.

Thực hiện các công việc tính toán và phân tích để có cái nhìn rõ hơn về bộ dữ liệu. Từ đó ta xử lý và chuẩn bị một bộ dữ liệu tốt cho mô hình chạy.

Xây dựng một mô hình tốt để dự đoán nhiệt độ, độ ẩm và tình trạng thời tiết hiện tại. Nhóm sẽ sử dụng mô hình LSTM để dự đoán các trường cần thiết, sau đó kết quả đầu ra sẽ được đưa qua thêm một mô hình (category classification) để phân loại thuộc tình trạng thời tiết nào .

Nhóm sẽ xây dựng một hệ thống để tự động thu thập dữ liệu cần thiết và huấn luyện lại mô hình để tăng thêm độ chính xác cho các phán đoán.

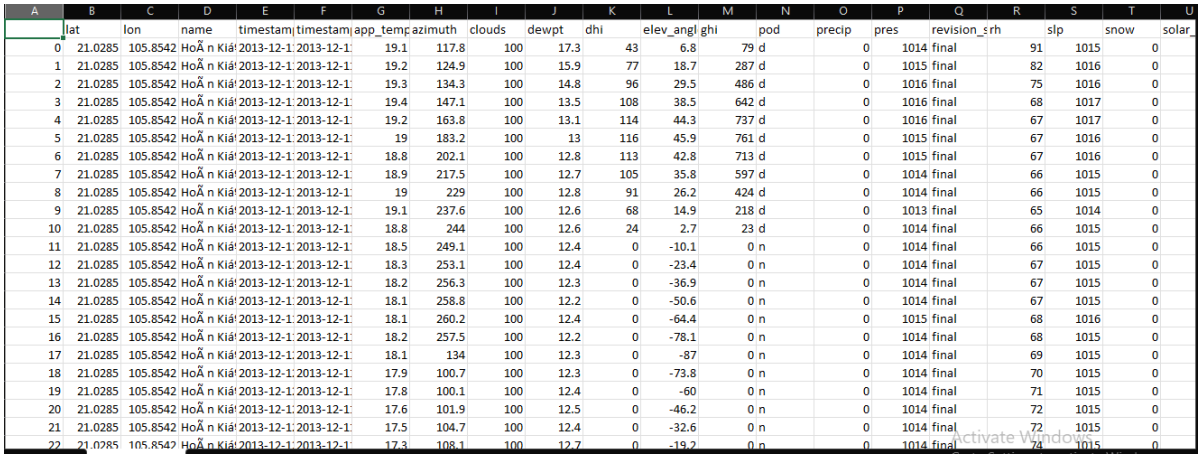
CHƯƠNG 3: THU THẬP DỮ LIỆU

3.1 Chọn nguồn thu thập dữ liệu

Nhóm sẽ chọn bộ dữ liệu được lấy từ trang web Weatherbit. Weatherbit cung cấp người dùng thông tin chi tiết về thời tiết hiện tại, thời tiết dài hạn cũng như có bộ database chứa thông tin thời tiết ở quá khứ của các thành phố. Do nhóm sử dụng dịch vụ miễn phí của trang web, nên việc crawl sẽ bị giới hạn ở 1500 request /1 ngày.

```
"data":[
   {
      "rh":32,
      "wind_spd":6.7,
      "wind_gust_spd": 9.4,
      "slp":1020.3,
      "h_angle":15,
      "azimuth":25,
      "dewpt":-7.5,
      "snow":0,
      "uv":0,
      "wind_dir":220,
      "weather":{
         "icon":"c01n",
         "code":"800",
         "description":"Clear sky"
      },
      "pod":"n",
      "vis":1.5,
      "precip":0,
      "elev_angle":-33,
      "ts":1483232400,
      "pres":1004.7,
      "datetime":"2018-05-01:06",
      "timestamp_utc":"2015-05-01T06:00:00",
      "timestamp_local":"2015-05-01T02:00:00",
      "revision_status":"final",
      "temp":8.3,
      "dhi":15,
      "dni":240.23,
      "ghi":450.9,
      "solar_rad":445.85,
      "clouds":0
   }, ...
],
"city_name":"Raleigh",
"city_id":"4487042"
}
```


Nhóm em đã lấy dữ liệu thời tiết của Hà Nội trong 10 năm gần đây, nhằm có thể có cái nhìn rõ nhất về dữ liệu, từ đó có thể xử lý chúng tốt hơn.



	lat	lon	name	timestam	timestam	app_temp	azimuth	clouds	dewpt	dhi	elev_angl	ghi	pod	precip	pres	revision_s	rh	slp	snow	solar_
0	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	19.1	117.8	100	17.3	43	6.8	79	d	0	1014	final	91	1015	0	
1	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	19.2	124.9	100	15.9	77	18.7	287	d	0	1015	final	82	1016	0	
2	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	19.3	134.3	100	14.8	96	29.5	486	d	0	1016	final	75	1016	0	
3	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	19.4	147.1	100	13.5	108	38.5	642	d	0	1016	final	68	1017	0	
4	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	19.2	163.8	100	13.1	114	44.3	737	d	0	1016	final	67	1017	0	
5	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	19	183.2	100	13	116	45.9	761	d	0	1015	final	67	1016	0	
6	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	18.8	202.1	100	12.8	113	42.8	713	d	0	1015	final	67	1016	0	
7	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	18.9	217.5	100	12.7	105	35.8	597	d	0	1014	final	66	1015	0	
8	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	19	229	100	12.8	91	26.2	424	d	0	1014	final	66	1015	0	
9	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	19.1	237.6	100	12.6	68	14.9	218	d	0	1013	final	65	1014	0	
10	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	18.8	244	100	12.6	24	2.7	23	d	0	1014	final	66	1015	0	
11	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	18.5	249.1	100	12.4	0	-10.1	0	n	0	1014	final	66	1015	0	
12	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	18.3	253.1	100	12.4	0	-23.4	0	n	0	1014	final	67	1015	0	
13	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	18.2	256.3	100	12.3	0	-36.9	0	n	0	1014	final	67	1015	0	
14	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	18.1	258.8	100	12.2	0	-50.6	0	n	0	1014	final	67	1015	0	
15	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	18.1	260.2	100	12.4	0	-64.4	0	n	0	1015	final	68	1016	0	
16	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	18.2	257.5	100	12.2	0	-78.1	0	n	0	1014	final	68	1015	0	
17	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	18.1	134	100	12.3	0	-87	0	n	0	1014	final	69	1015	0	
18	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	17.9	100.7	100	12.3	0	-73.8	0	n	0	1014	final	70	1015	0	
19	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	17.8	100.1	100	12.4	0	-60	0	n	0	1014	final	71	1015	0	
20	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	17.6	101.9	100	12.5	0	-46.2	0	n	0	1014	final	72	1015	0	
21	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	17.5	104.7	100	12.4	0	-32.6	0	n	0	1014	final	72	1015	0	
22	21.0285	105.8542	HoÃ n Kiáº	2013-12-1	2013-12-1	17.3	108.1	100	12.7	0	-19.2	0	n	0	1014	final	74	1015	0	

CHƯƠNG 4: TIỀN XỬ LÝ DỮ LIỆU

4.1 Tìm hiểu các trường dữ liệu

Dữ liệu thu thập được gồm 32 cột, trong đó có 29 cột đáng chú ý:

1. Lat : Vĩ độ
2. Lon: Kinh độ
3. name: Địa điểm
4. Timestamp_local: Thời gian tại địa điểm
5. Timestamp_UTC: Thời gian theo UTC
6. app_temp: nhiệt độ biểu kiến (HI, chỉ số nóng bức cảm nhận)
7. azimuth: Góc phương vị (góc giữa ảnh mặt trời chiếu xuống so với hướng bắc)
8. clouds: Lượng mây phủ
9. dewpt: nhiệt độ hóa sương
10. dhi: lượng bức xạ mặt trời sau khi tán xạ trên 1 diện tích mặt đất
11. elev_angle: góc ngẩng (giữa đường chân trời và đường từ mắt đến mặt trời)
12. ghi: tổng lượng bức xạ của mặt trời lên phương ngang trên 1 diện tích mặt đất
13. pod: phần của ngày (ở đây được ghi chép với 2 giá trị d (ngày) và n (đêm))
14. precip: lượng mưa (giáng thủy)
15. pres: áp suất không khí
16. revision_status: tình trạng kiểm tra dự báo
17. rh: độ ẩm tương đối
18. slp: áp suất mực nước biển
19. snow: lượng tuyết
20. solar_rad: lượng bức xạ mặt trời
21. temp: Nhiệt độ đo được
22. ts: timestamp UNIX
23. vis: tầm nhìn xa
24. uv: lượng uv
25. weather _code: mã thời tiết

26. weather_description: thời tiết

27. wind_dir: hướng gió

28. wind_gust: độ gió giật

29. wind_spd: tốc độ gió

Những đặc trưng trên có đặc điểm thống kê được mô tả ở bảng sau (Cụ thể hơn xin xem ở notebook):

	lat	lon	app_temp	azimuth	clouds	dewpt	dhi	...	snow	solar_rad	temp	ts	uv	vis
8.764800e+04	87648.0000	87648.000000	87648.000000	87648.000000	87648.000000	87648.000000	87648.000000	...	87648.0	87648.000000	87648.000000	8.764800e+04	87648.000000	87183.000000
2.102850e+01	105.8542	27.611620	179.745801	71.154790	20.464375	47.459292	...	0.0	163.491021	24.906150	1.544485e+09	1.273389	9.950873	
7.105468e-15	0.0000	8.407815	97.828684	28.109117	5.759539	51.809806	...	0.0	243.417183	5.633962	9.108699e+07	1.856148	2.480056	
2.102850e+01	105.8542	3.000000	0.000000	0.000000	-6.900000	0.000000	...	0.0	0.000000	5.900000	1.386720e+09	0.000000	0.000000	
2.102850e+01	105.8542	21.100000	89.300000	50.000000	17.000000	0.000000	...	0.0	0.000000	20.800000	1.465602e+09	0.000000	10.000000	
2.102850e+01	105.8542	26.500000	180.000000	75.000000	22.300000	13.000000	...	0.0	15.000000	25.700000	1.544485e+09	0.600000	10.000000	
2.102850e+01	105.8542	34.300000	270.500000	100.000000	25.000000	107.000000	...	0.0	236.000000	28.900000	1.623367e+09	2.000000	10.000000	
2.102850e+01	105.8542	53.100000	360.000000	100.000000	29.700000	128.000000	...	0.0	1061.000000	42.100000	1.702249e+09	12.100000	16.000000	

4.2 Tiền xử lý dữ liệu

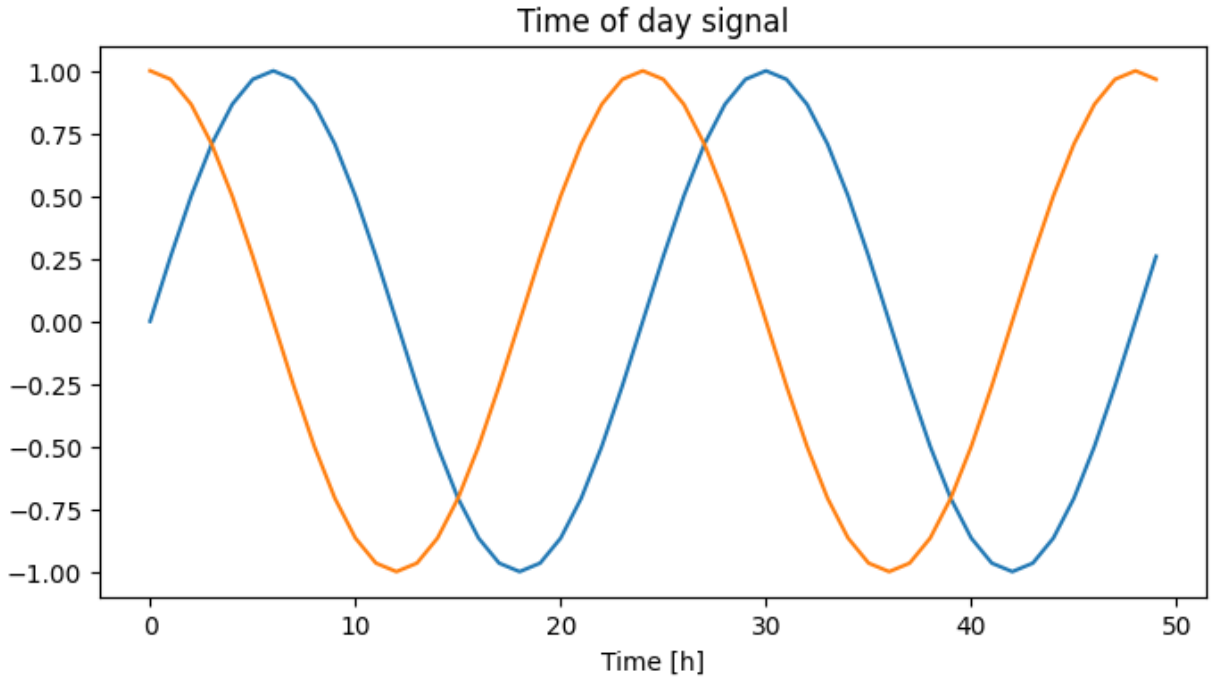
1. Loại bỏ các trường không có giá trị phân tích hoặc giá trị mẫu gặp hiện tượng imbalance, khuyết thiếu nhiều: 'timestamp_local', 'ts', 'name', 'pod', 'weather_code', 'revision_status', 'weather_description', 'lat', 'lon', 'snow', 'precip', 'vis'

2. Các đại lượng mang giá trị góc thường không tạo ra các đầu vào tốt. Tiến hành chuyển các giá trị góc về các giá trị sin và cos của góc này.

```
azimuth = df_test.pop('azimuth')
elev_angle = df_test.pop('elev_angle')
wind_dir = df_test.pop('wind_dir')

df_test['azimuth_sin'] = np.sin(azimuth * np.pi / 180)
df_test['azimuth_cos'] = np.cos(azimuth * np.pi / 180)
df_test['elev_angle_sin'] = np.sin(elev_angle * np.pi / 180)
df_test['elev_angle_cos'] = np.cos(elev_angle * np.pi / 180)
df_test['wind_dir_sin'] = np.sin(wind_dir * np.pi / 180)
df_test['wind_dir_cos'] = np.cos(wind_dir * np.pi / 180)
```

3. Chuyển thời gian định dạng UTC thành định dạng Unix để dễ dàng xử lí bước tiếp theo. Từ thời gian định dạng Unix thu được ta chuyển về các giá trị sin và cos tính theo ngày (các giá trị này mang nhiều ý nghĩa để xác định thời gian trong 1 ngày)

```
date_time = pd.to_datetime(df_test.pop('timestamp_utc'), format='%Y-%m-%dT%H:%M:%S')
timestamp_s = date_time.map(pd.Timestamp.timestamp)

day = 24*60*60
df_test['day_sin'] = np.sin(timestamp_s * (2 * np.pi / day))
df_test['day_cos'] = np.cos(timestamp_s * (2 * np.pi / day))

plt.plot(np.array(df_test['day_sin'])[:50])
plt.plot(np.array(df_test['day_cos'])[:50])
plt.xlabel('Time [h]')
plt.title('Time of day signal')
```



4. Chia tập dữ liệu thành các tập train, validation, test theo tỉ lệ 7:2:1

5. Chuẩn hóa Z-score thường được áp dụng trong quá trình tiền xử lý dữ liệu khi làm việc với các mô hình máy học. Điều này có thể giúp tăng tốc quá trình học của mô hình và cải thiện hiệu suất. Phương pháp này nhằm chuyển đổi một tập dữ liệu thành một phân phối có trung bình (mean) bằng 0 và độ lệch chuẩn (standard deviation) bằng 1. Tiến hành chuẩn hóa z_score trên tập train, sau đó áp dụng trên tập validation và tập test.

```
train_mean = train_df.mean()
train_std = train_df.std()

train_df = (train_df - train_mean) / train_std
val_df = (val_df - train_mean) / train_std
test_df = (test_df - train_mean) / train_std
```

4.3 Feature Engineering

Những nội dung sau sử dụng cho phần EDA.

1. Loại bỏ một số trường không cần thiết

- Loại bỏ một số trường không có giá trị phân tích: lat, lon, hai giá trị này phụ thuộc vào địa lý, nhưng ta chỉ sử dụng 1 địa điểm duy nhất.

- Loại bỏ trường không có giá trị khí tượng ở Việt Nam: snow, đặc trưng này có giá trị bằng 0 trên tất cả các mẫu

2. Chuyển đổi định dạng thời gian

- Khi phân tích dữ liệu dạng thời gian, ta sẽ phải chuyển dữ liệu này thành dạng dd/mm/yyyy. Ta sử dụng thư viện timedelta và gọi phương thức to_datetime để tạo column mới “Date”.

```
[8]  import datetime as dt
     from datetime import timedelta
```

```
[9]  data["Date"]=pd.to_datetime(data['timestamp_local'])
```

3. Loại bỏ một số mẫu có giá trị null ở ‘vis’

```
data.dropna(subset=['vis'], inplace=True)
data.isnull().sum()
```

```
lat                     0
lon                     0
timestamp_local         0
app_temp                0
azimuth                 0
clouds                  0
dewpt                   0
dhi                     0
elev_angle              0
ghi                     0
precip                  0
pres                    0
rh                      0
slp                     0
snow                    0
solar_rad               0
temp                    0
uv                      0
vis                     0
weather_code            0
weather_description     0
wind_dir                0
wind_gust_spd           0
wind_spd                0
dtype: int64
```

4. Lấy mẫu trung bình theo tháng

Kỹ thuật này thường áp dụng cho việc phân tích dữ liệu theo thời gian. Lấy mẫu trung bình theo tháng/ năm giúp ta nhận biết được xu hướng của đặc trưng trong bài toán và loại bỏ các nhiễu khi phân tích.

```
#Mối quan hệ giữa app_temp và pres theo hồi quy
#lấy mẫu theo tháng
df_column = ['Date','temp', 'pres']
df_monthly_mean = data[df_column].resample("MS", on='Date').mean()

df_monthly_mean
```

Date	temp	pres
2013-12-01	13.848290	1019.832998
2014-01-01	17.196102	1018.450269
2014-02-01	17.052530	1014.157738
2014-03-01	19.806586	1012.899194
2014-04-01	24.987917	1008.540278
...	...	...
2023-08-01	29.375134	1001.825269
2023-09-01	28.512917	1006.183333
2023-10-01	27.394758	1012.685484
2023-11-01	23.685417	1016.340278
2023-12-01	20.651012	1015.659919

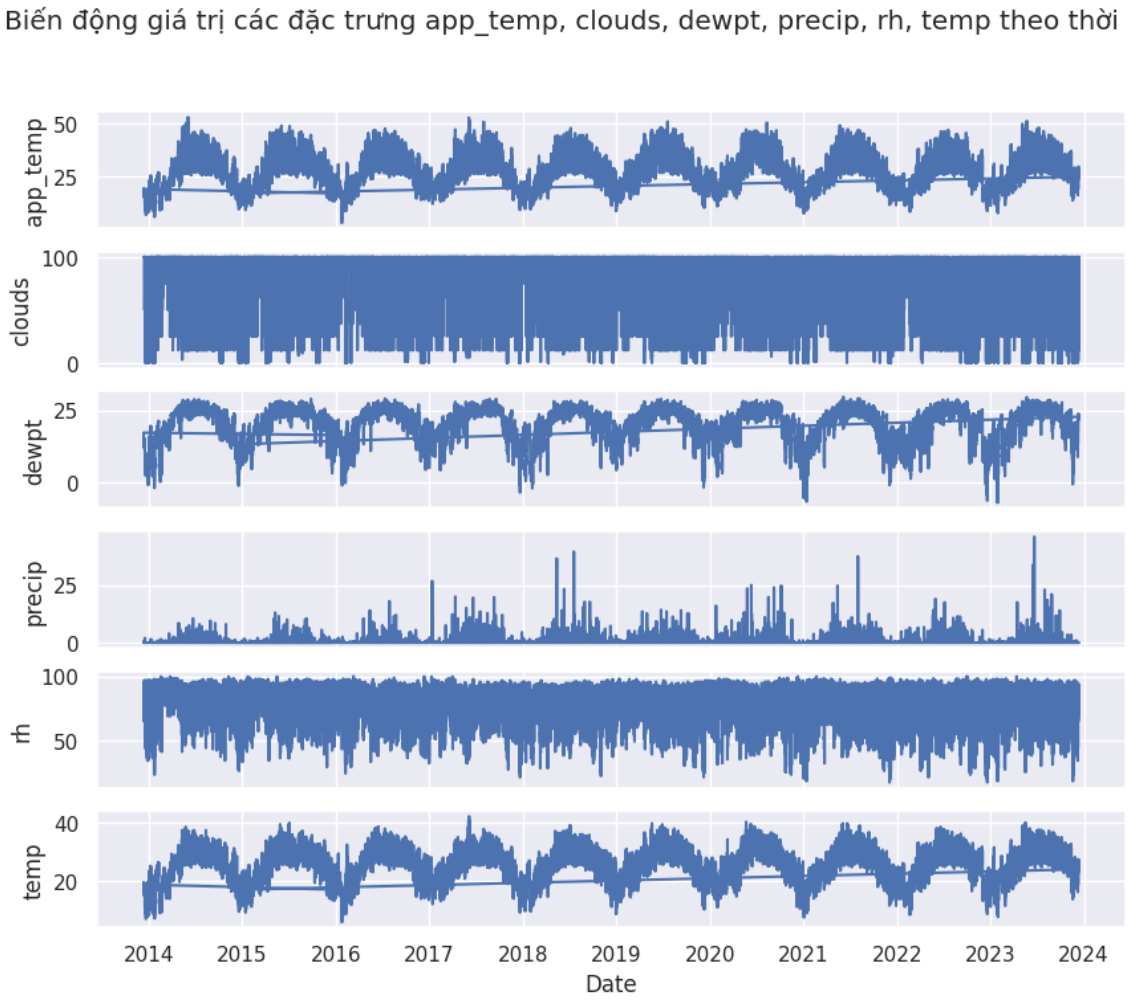
121 rows × 2 columns

CHƯƠNG 5: PHÂN TÍCH DỮ LIỆU

5.1 Trực quan hóa dữ liệu

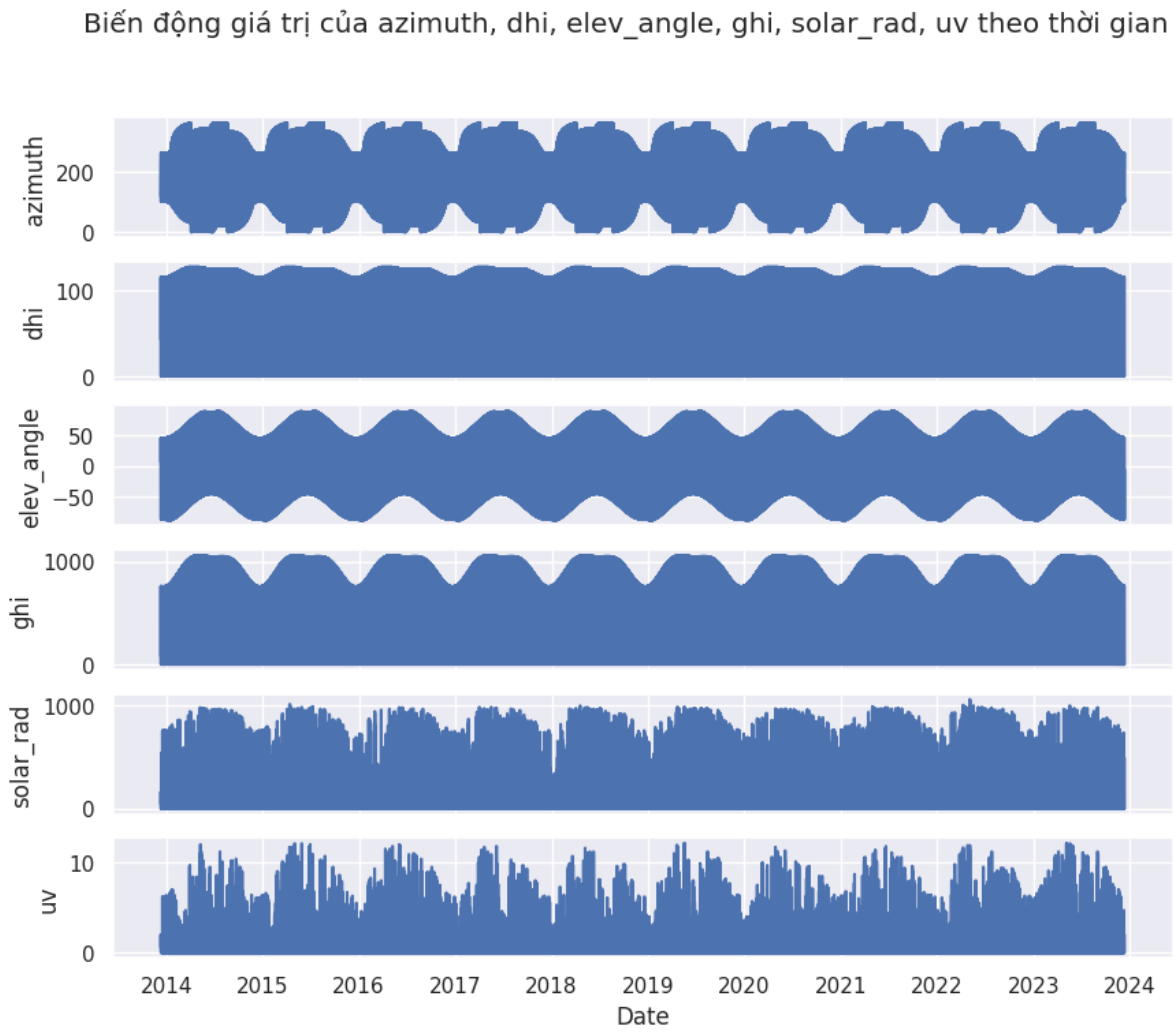
1. Phân tích về độ biến động của từng đặc trưng theo thời gian

Cách phân tích này dùng để phân loại các nhóm đặc trưng ảnh hưởng theo địa lý; đặc trưng ảnh hưởng theo thời gian (ngày, tháng và năm)



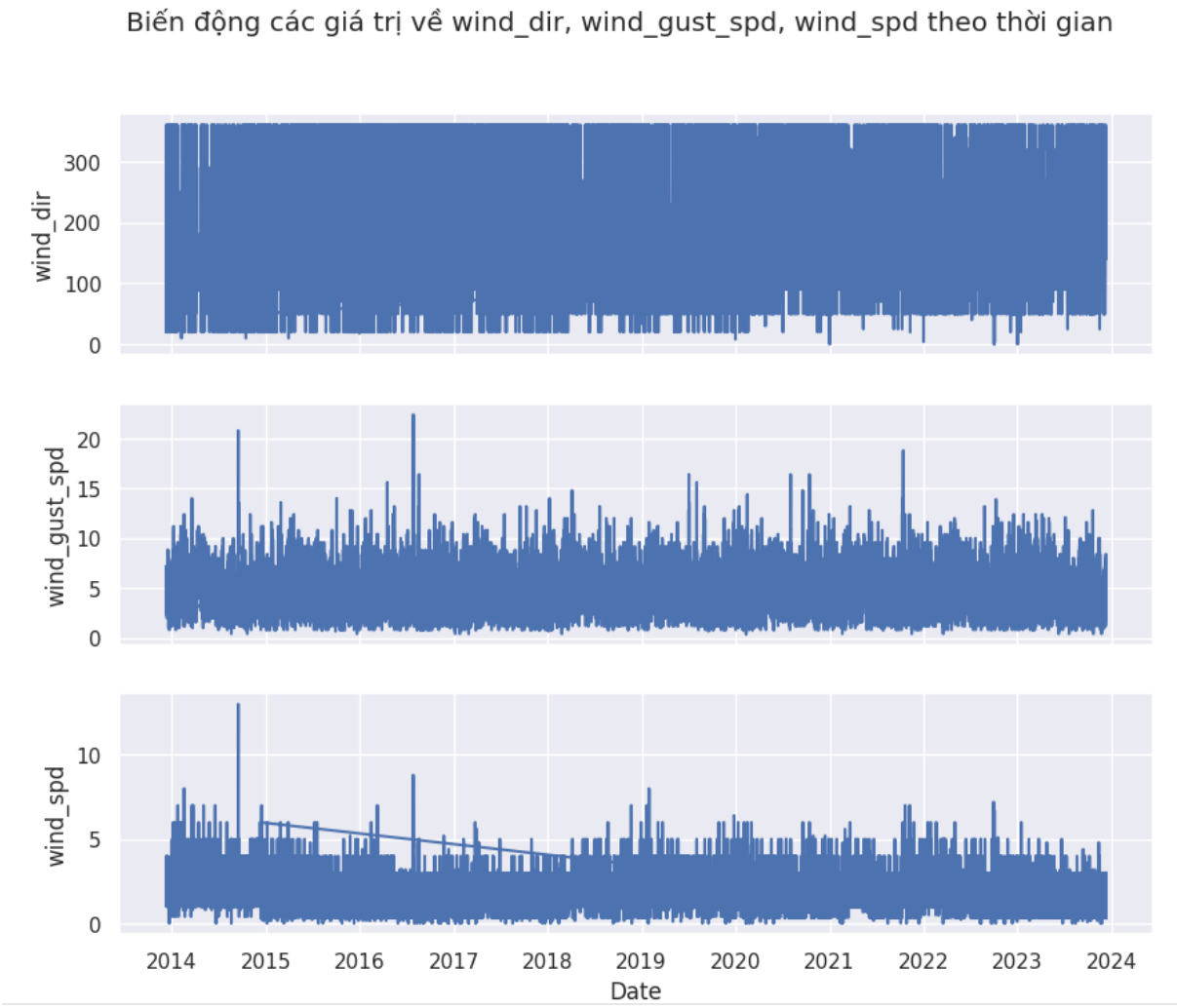
Hình 1: Biến động giá trị các đặc trưng app_temp, clouds, dewpt, precip, rh, temp theo thời gian

Quan sát: Những đặc trưng khí tượng như app_temp, temp, dewpt, percip, temp có xu hướng đối xứng. Đây là những đặc trưng diễn ra theo chu kỳ trong năm, do đó ít trơn mịn hơn. Với rh và clouds thì khá dày và biên độ đều, không theo xu hướng.



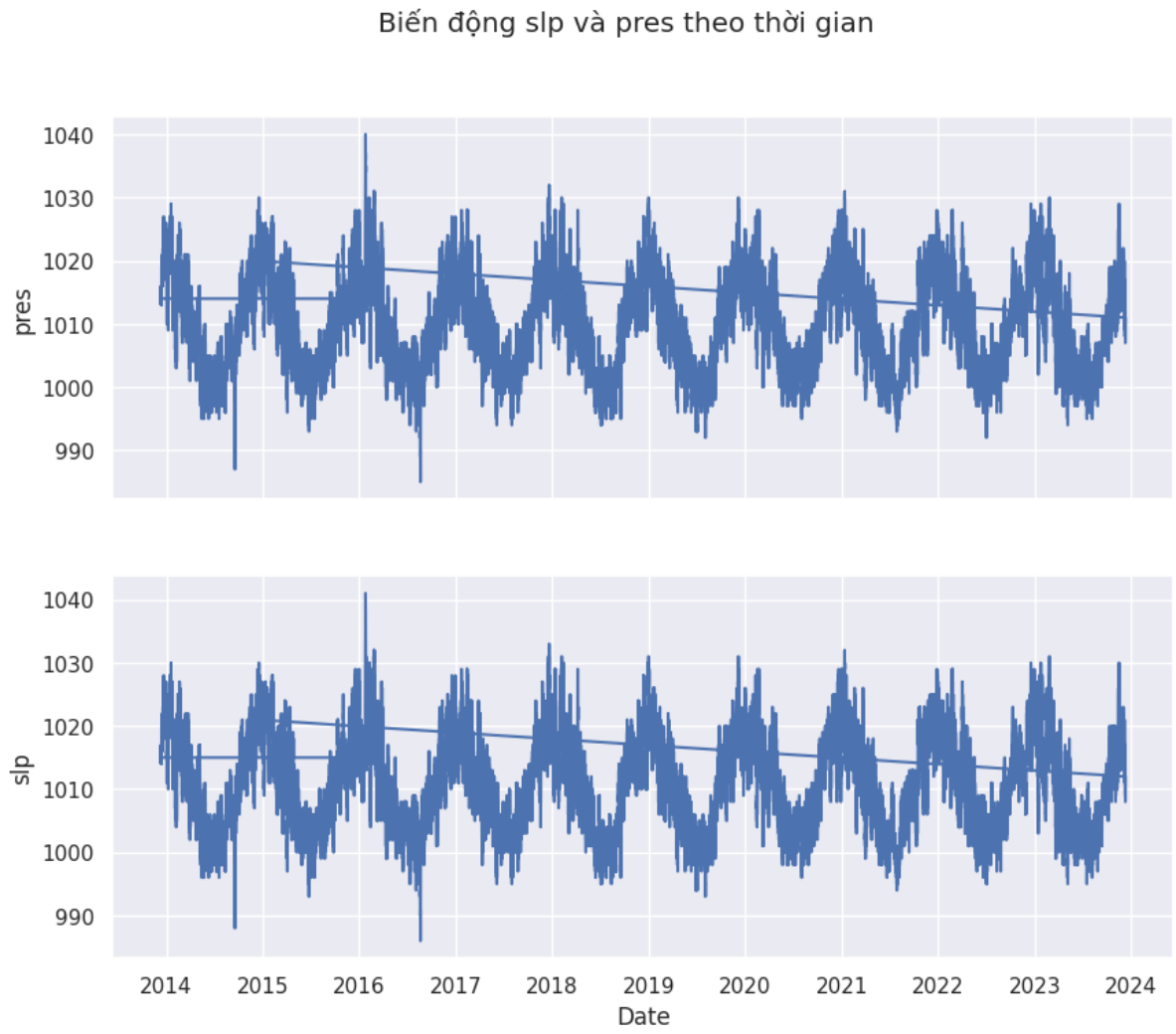
Hình 2: Biến động giá trị của azimuth, dhi, elev_angle, ghi, solar_rad, uv theo thời gian

Quan sát: Những đặc trưng khí tượng như app_temp, temp, dewpt, percip, temp có xu hướng đối xứng. Đây là những đặc trưng diễn ra theo chu kỳ trong năm, do đó ít trơn mịn hơn. Với rh và clouds thì khá dày và biên độ đều, không theo xu hướng.



Hình 3: Biến động các giá trị về wind_dir, wind_gust_spd, wind_spd theo thời gian

Quan sát: Tương tự những quan sát trên, wind_spd, wind_gust_spd không phụ thuộc theo thời gian nên hình ảnh không có mô tả xu hướng đáng kể

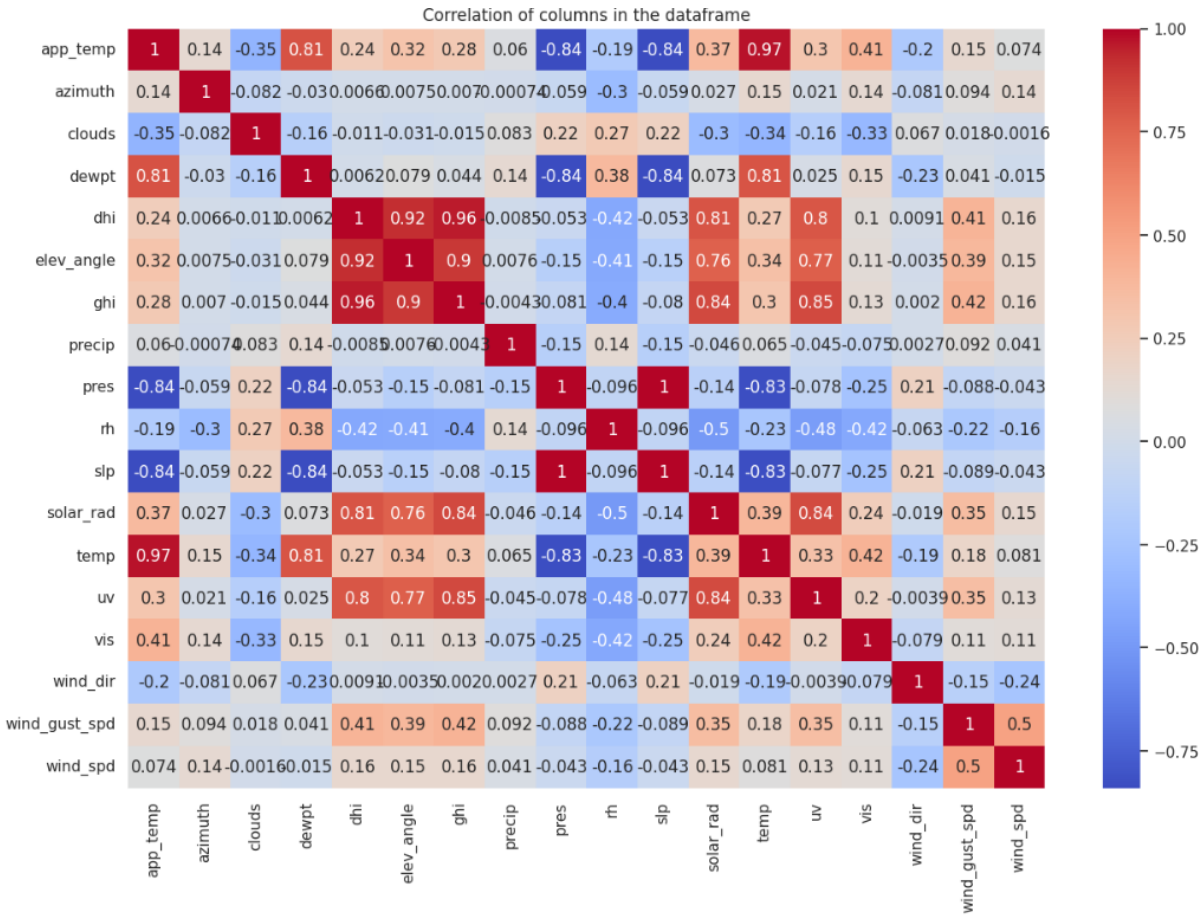


Hình 4: Biến động slp và pres theo thời gian

Quan sát: pres và slp là những đặc trưng phụ thuộc thời gian

- Tóm lại: Những đặc trưng theo thời gian thể hiện xu hướng thay đổi theo thời gian và có tính đối xứng, biên độ thay đổi không lớn.

2. Phân tích tương quan đặc trưng dựa trên heatmap và kiểm chứng độ quan trọng của đặc trưng



Hình 5: Biểu đồ nhiệt

Các đặc trưng địa lý diễn ra theo ngày như quan sát trước đó: 'azimuth', 'dhi', 'elev_angle', 'ghi', 'solar_rad', 'uv' thường có ít mối quan hệ đến các đặc trưng khác. Lưu ý nhóm chỉ số 'dhi', 'elev_angle', 'ghi', 'solar_rad', 'uv' liên quan với nhau bởi khái niệm lượng bức xạ nên phụ thuộc rõ rệt (tỉ lệ thuận)

Các cặp đặc trưng tỉ lệ nghịch với nhau lớn <-0,45:

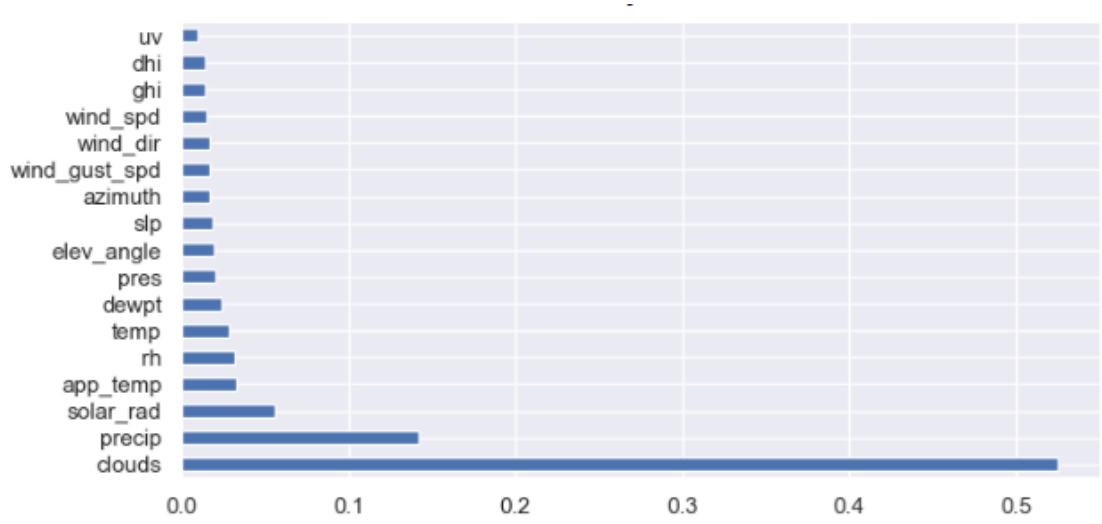
- app_temp & pres, app_temp & slp
- dewpt & pres, dewpt & slp
- pres & temp, slp & temp
- rh & solar_rad, rh & uv

Các đặc trưng tỉ lệ thuận với nhau > 0,45:

- app_temp &dewpt, app_temp &temp
- dewpt &temp
- dhi & elev_angle, dhi &ghi, dhi & solar_rad, dhi & uv,...

Từ đó, một số đặc trưng ít liên quan đến như 'clouds', 'precip', 'rh' trở nên thành những đặc trưng quan trọng vì tính độc lập của chúng.

Ta kiểm nghiệm độ quan trọng của các đặc trưng sử dụng tree_based classifiers



Hình 6: Độ quan trọng các đặc trưng thời tiết

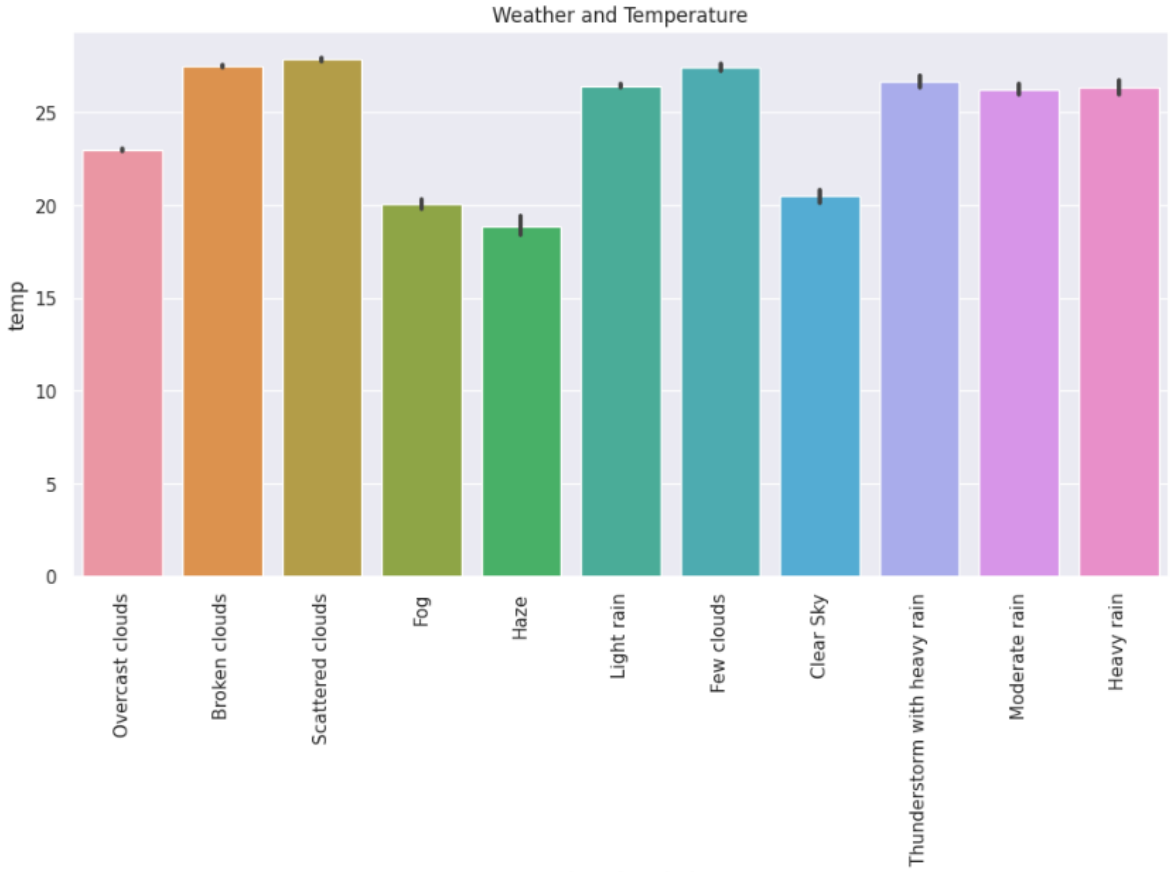
Ta thấy xếp hạng độ quan trọng từ clouds> precip>solar_rad>.... Biểu đồ nhiệt giúp ta chỉ ra được những đặc trưng quan trọng một cách trực quan.

3. Phân tích tương quan nhãn dữ liệu và các đặc trưng

	Weather	Count
0	Overcast clouds	44666
1	Broken clouds	18580
2	Scattered clouds	9764
3	Light rain	7021
4	Few clouds	3828
5	Clear Sky	1435
6	Fog	939
7	Moderate rain	674
8	Thunderstorm with heavy rain	274
9	Heavy rain	250
10	Haze	217

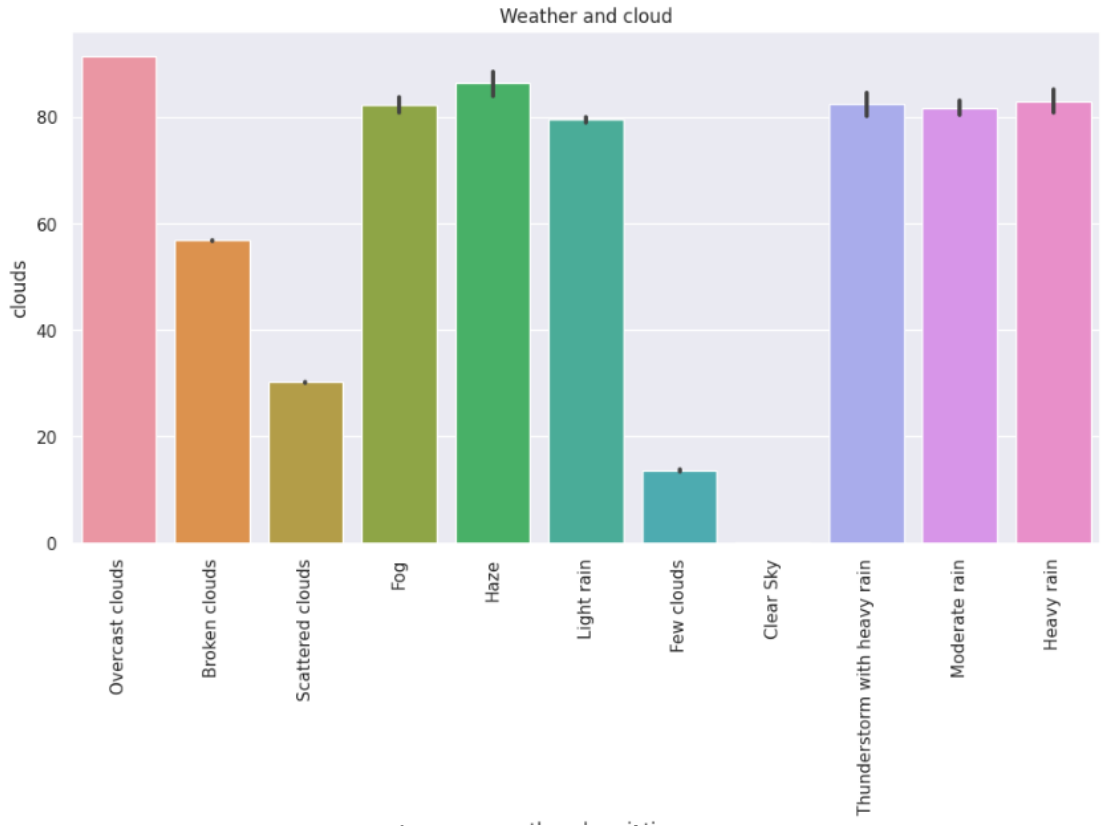
Hình 7: Số lượng từng nhãn thời tiết.

Chúng ta có tận 11 nhãn thời tiết, số lượng mẫu cho từng nhãn rất mất cân bằng (lớp thiểu số chiếm 0,24%<1%). Thời tiết ở nơi này đa phần liên quan về mây.



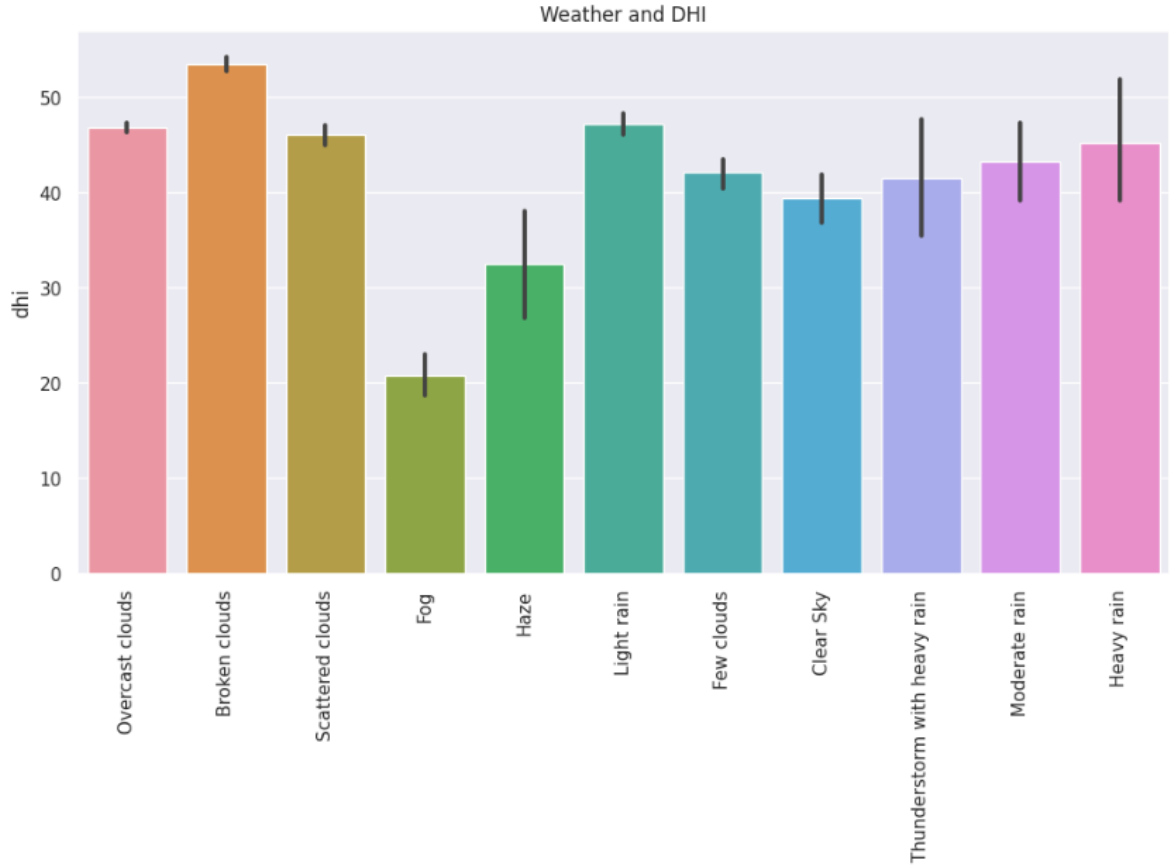
Hình 8: Thể hiện nhãn thời tiết theo giá trị nhiệt độ

Quan sát: Đối với đặc trưng temp, độ lệch chuẩn trên toàn bộ mẫu của từng nhãn không lớn, riêng với nhãn Haze có độ lệch chuẩn lớn nhất, chứng tỏ các giá trị của temp sai khác nhiều với từng mẫu nhãn này.



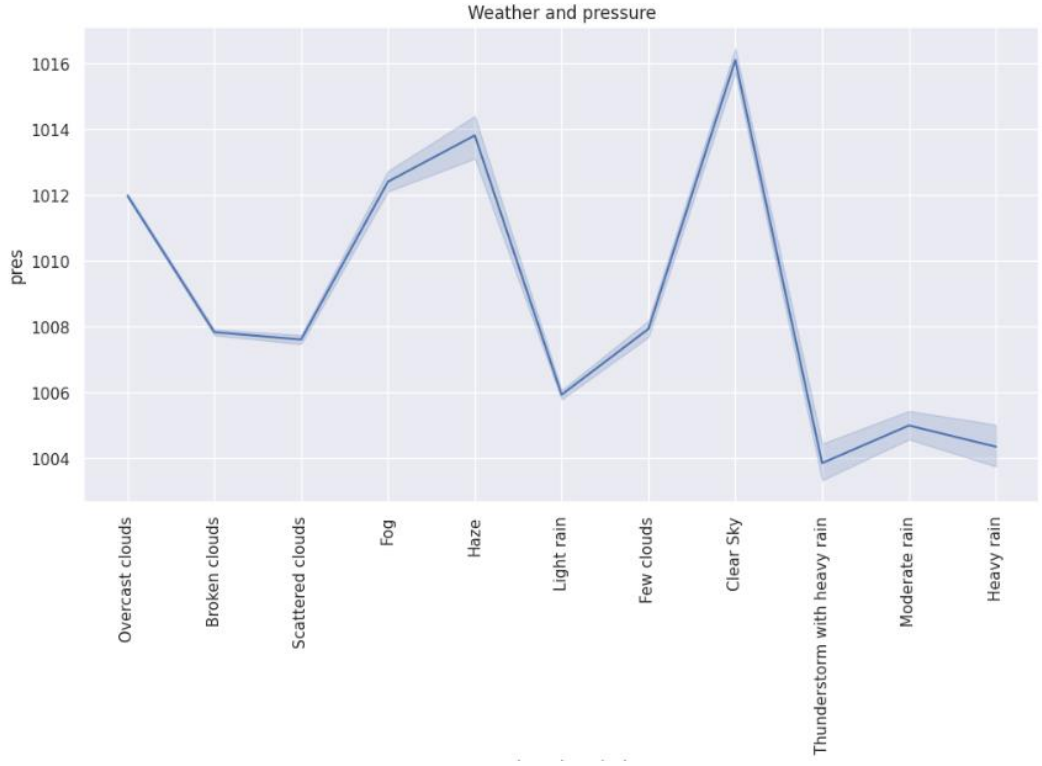
Hình 9: Thể hiện nhãn thời tiết theo giá trị mây phủ

Quan sát: Đặc trưng clouds cũng tương tự, nhưng đặc biệt với xu hướng lượng mây tăng dần thì số lượng mẫu cho từng nhãn liên quan về lượng mây cũng tăng. Với nhãn thời tiết trời quang (clear sky) thì clouds là 0.



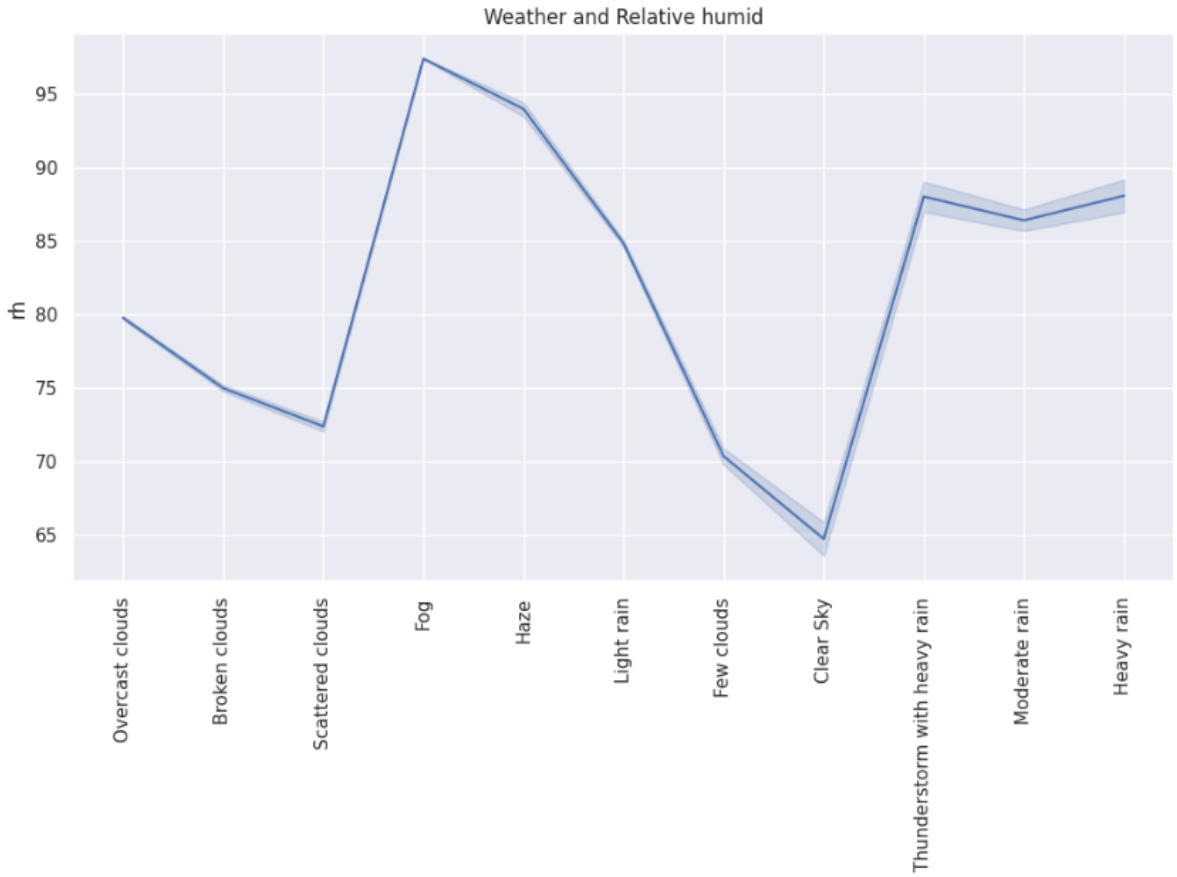
Hình 10: Thể hiện nhãn thời tiết theo giá trị mây phủ bức xạ tán xạ

Quan sát: Đối với đặc trưng DHI, độ lệch chuẩn trên các nhãn dữ liệu khá lớn, đặc biệt lớn với Haze, và các nhãn liên quan đến rain.



Hình 11: Thể hiện nhãn thời tiết theo giá trị áp suất không khí

Khi trời quang (clear sky) thì áp suất thể hiện là cao nhất (1016). Các nhãn thời tiết có độ lệch giá trị nhiều nhất (Haze, Thunderstorm with heavy rain, Moderate rain, Heavy rain)



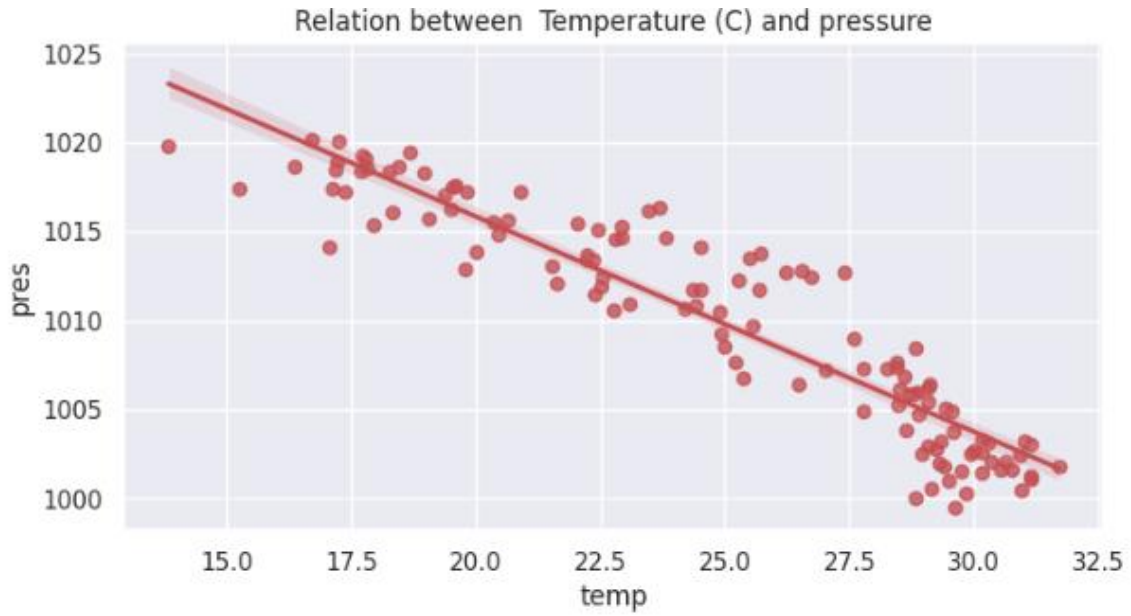
Hình 12: Thể hiện nhãn thời tiết theo độ ẩm tương đối

Ngược lại, khi trời quang thì độ ẩm tương đối thấp (65) và trời sương mù có độ ẩm tương đối lớn nhất (~93 đến 94). Xu hướng: thời tiết từ mây đến mưa đến sương mù thì độ ẩm tăng theo.

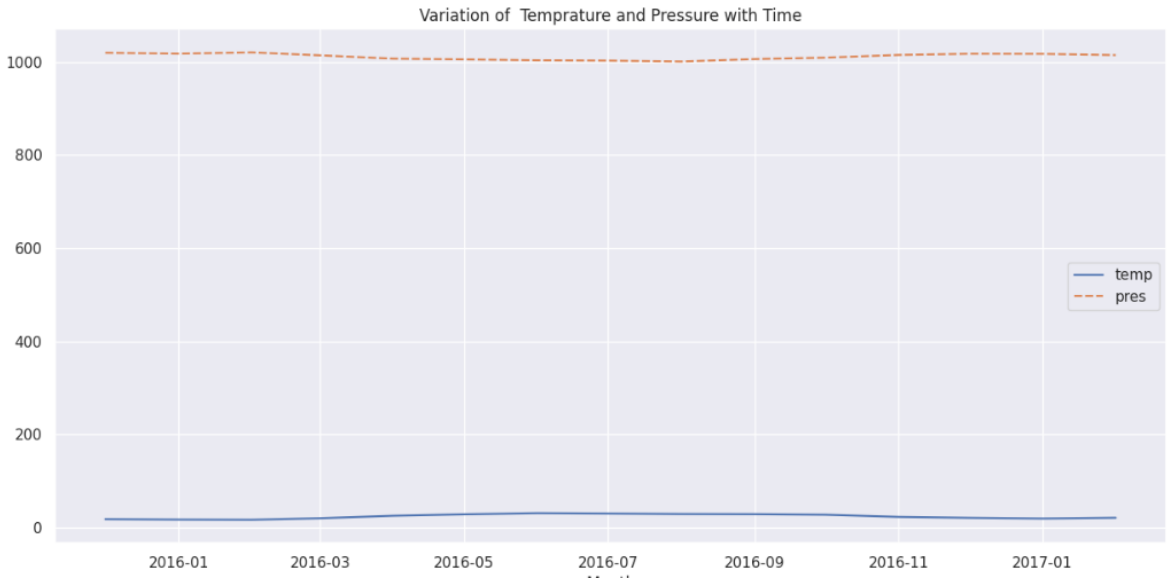
- Tóm lại:
- Các đặc trưng ảnh hưởng theo ngày làm cho mỗi nhãn lớp có độ lệch chuẩn lớn, khó để nhận biết được sau này.
- Các đặc trưng quan trọng như clouds, rh, temp,... thể hiện cụ thể những nhãn lớp cụ thể hơn.

4. Phân tích theo cặp đặc trưng dựa trên hồi quy

Các thao tác sau đây đều dựa trên việc lấy mẫu trung bình theo tháng

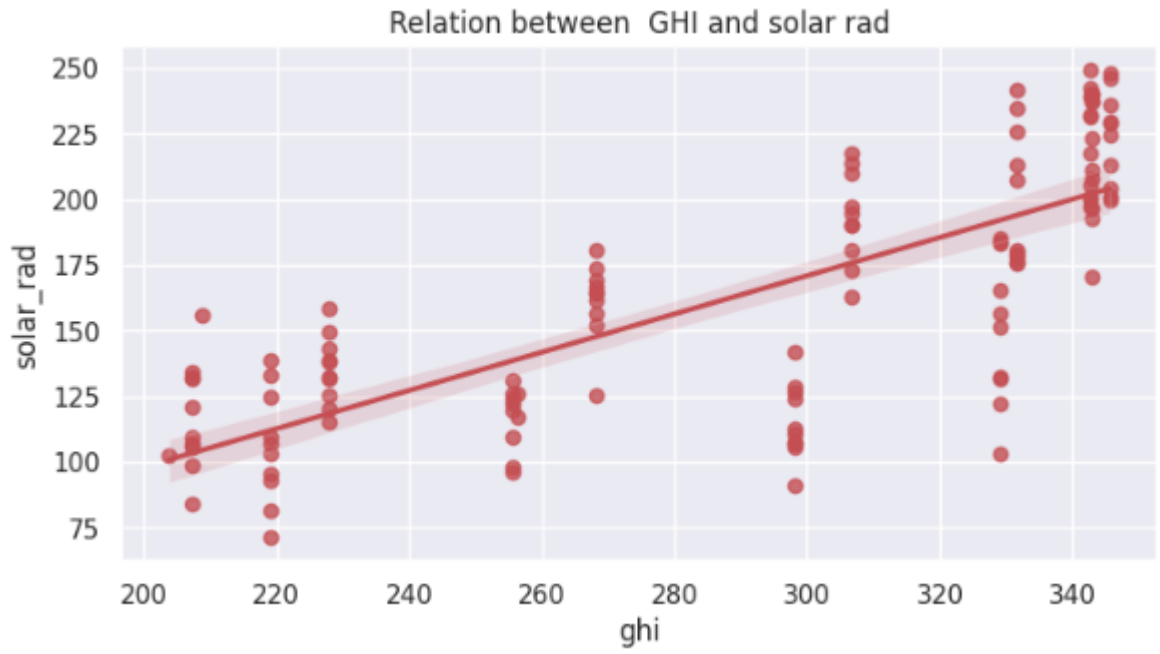


Hình 13: Mô tả quan hệ hồi quy giữa nhiệt độ và áp suất

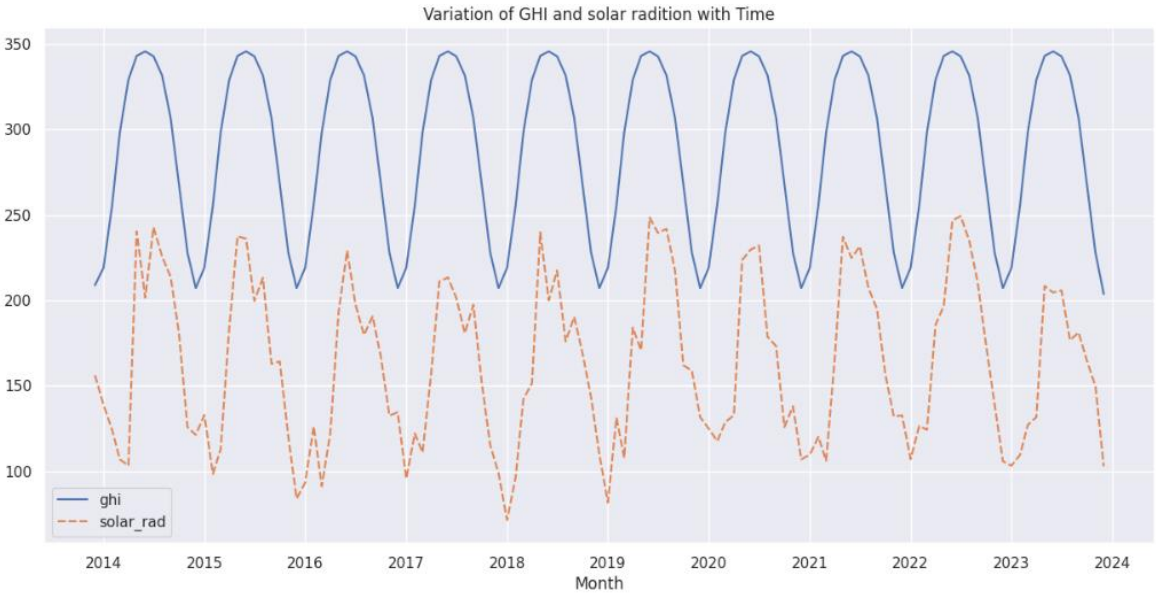


Hình 13: Mô tả quan hệ hồi quy giữa nhiệt độ và áp suất

Quan sát: Đây là mối quan hệ tỉ lệ nghịch. Càng nhiệt độ cao, áp suất KK càng giảm (khoảng tháng 5 đến tháng 11 năm 2016 thể hiện tương đối rõ). Một kiểm chứng khoa học: Khi nhiệt độ càng tăng thì phân tử nhận được năng lượng chuyển động ra xa khỏi nhau dần, sự va chạm lẫn nhau giảm đi khiến áp suất không khí giảm (ở 1 độ cao nhất định)

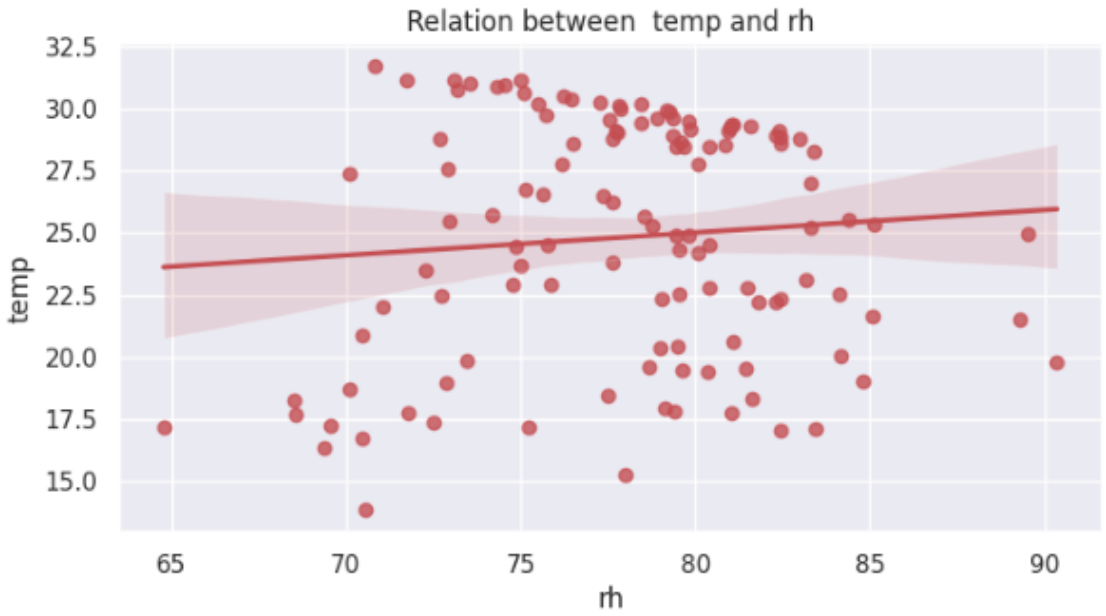


Hình 14: Mô tả quan hệ hồi quy giữa GHI và solar_rad

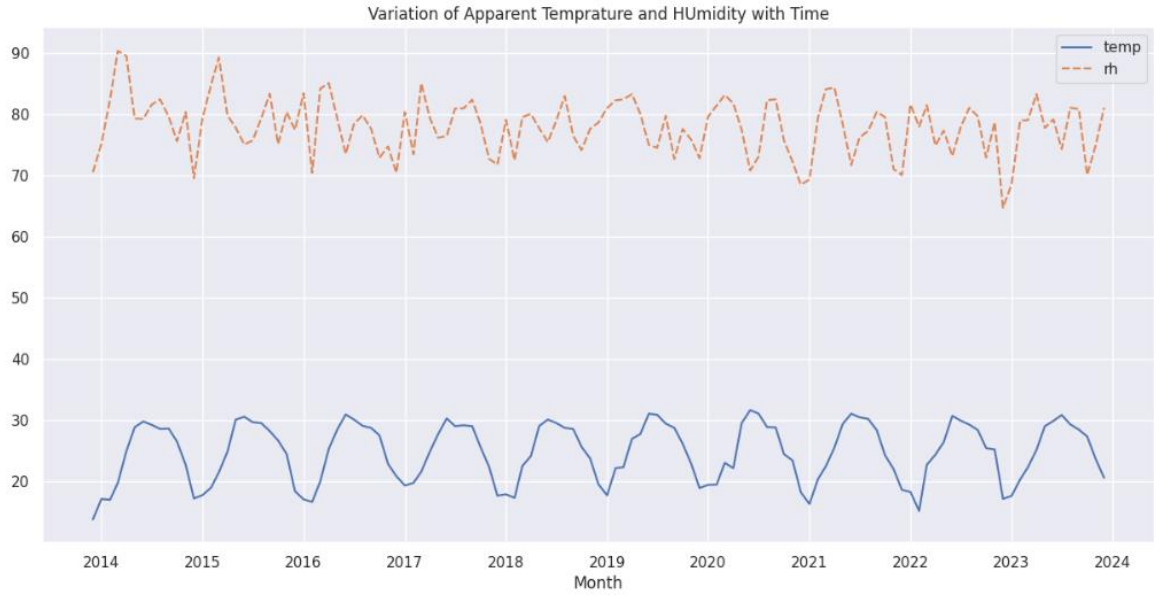


Hình 15: Mô tả trực quan về sự thay đổi giá trị GHI và solar_rad

Hai đặc trưng solar_rad và GHI có sự tương quan tỉ lệ thuận cao (heatmap: 0.84)



Hình 16: Mô tả quan hệ hồi quy giữa nhiệt độ và độ ẩm

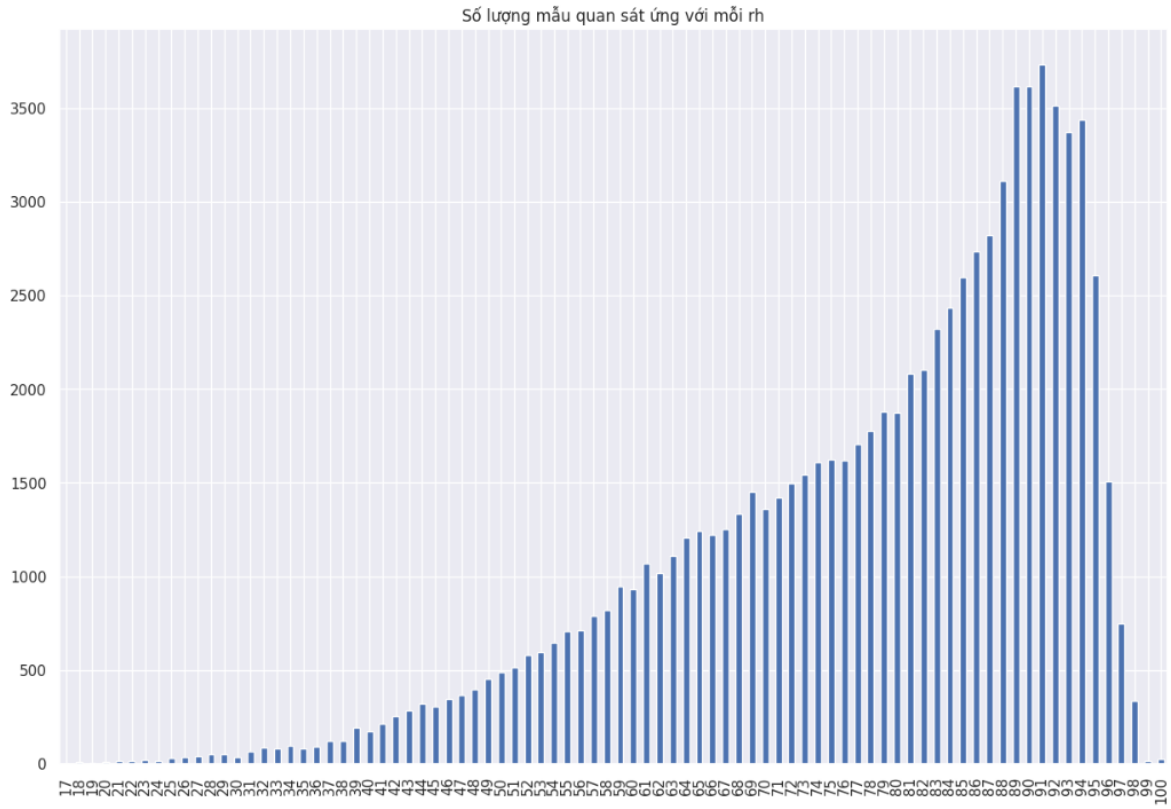


Hình 17: Mô tả trực quan về sự thay đổi giá trị nhiệt độ và độ ẩm

Hai đặc trưng thể hiện mối quan hệ hồi quy kém (heat map: -0.23), ta gần như không thể giải thích sự phụ thuộc lẫn nhau giữa 2 đặc trưng này

5. Phân tích ảnh hưởng đặc trưng lên phán đoán dựa trên kiểm định giả thiết

Ở đây, ta lấy đặc trưng rh làm ví dụ, ta muốn kiểm định xem độ ẩm (rh) có ảnh hưởng lên nhiệt độ hay không? Đầu tiên ta xem đặc trưng này có bao nhiêu giá trị và số lượng của từng giá trị.

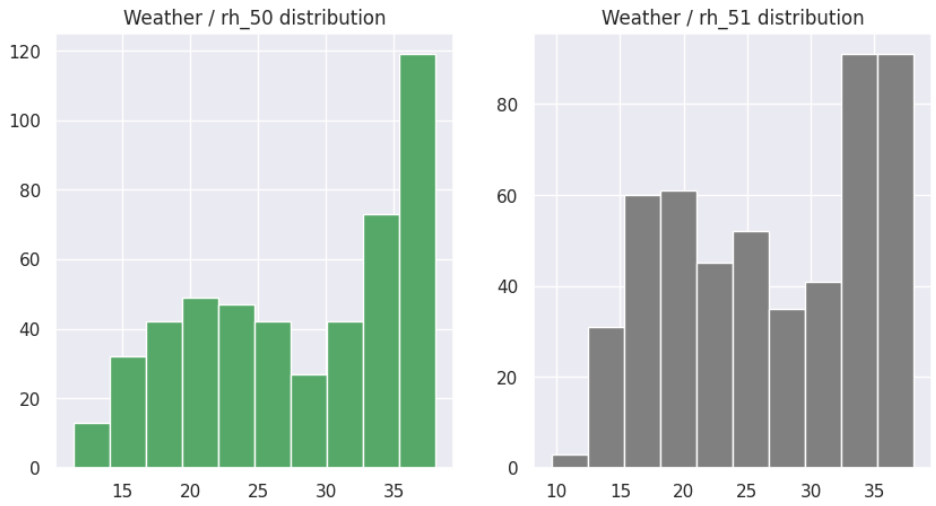


Hình 18: Số lượng mẫu quan sát ứng với mỗi giá trị độ ẩm.

Sau đó, ta kiểm tra xem 2 giá trị liên tiếp nhau có cùng phân phối chuẩn hay không đối với các giá trị nhiệt độ. Lưu ý dữ liệu ở dạng liên tục.

```
[ ]  ### Null hypothesis: dữ liệu tuân theo phân phối chuẩn ###
     ### Nếu pValue < 0.05 ==> phản bác null hypothesis ###
     from scipy import stats
     rh_50_dist = stats.shapiro(rh_50)
     rh_51_dist = stats.shapiro(rh_51)

     print('pvalue for rh_50 distribution: ', rh_50_dist[1])
     print('pvalue for rh_51 distribution: ', rh_51_dist[1])

     pvalue for rh_50 distribution:  9.965465708921991e-17
     pvalue for rh_51 distribution:  2.7548309300413746e-16
```

Rõ ràng, phân phối của 2 giá trị này khác phân phối chuẩn (với độ tin cậy 95%)



Hình 19: Minh họa phân phối của rh=50 và rh=51 đối với temp.

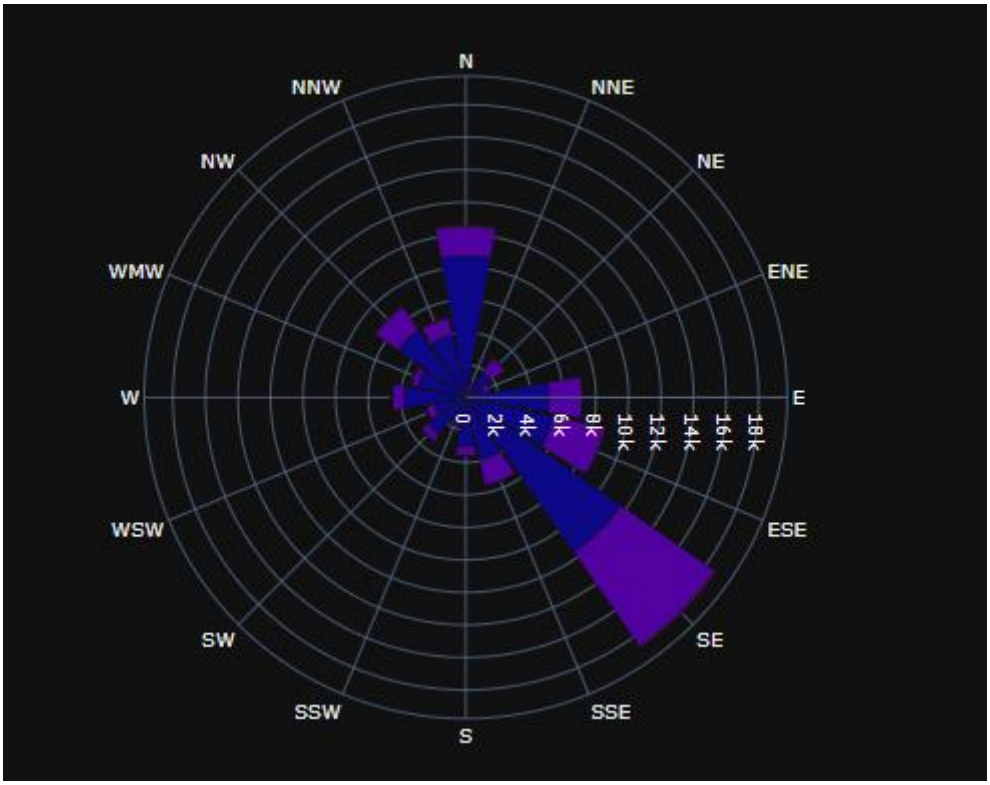
Không cùng phân phối chuẩn, ta sẽ kiểm tra với giả thuyết null: các giá trị rh không có cùng phân phối đối với nhiệt độ của kiểm thử Man Whitnet U .

```
#Null: 2 phân phối như nhau
different = stats.mannwhitneyu(rh_50, rh_51  , alternative='two-sided')


if different[1] < 0.05:
  print('Nhiệt độ không giống nhau về mặt thống kê với 2 giá trị rh 50 và 51 với độ tin cậy 0.05- Tức là độ ẩm có ảnh hưởng đến nhiệt độ')
else:
  print('Không thể kết luận nhiệt độ đối với 2 giá trị rh như trên là khác nhau với độ tin cậy 0.05.')
```

```
Nhiệt độ không giống nhau về mặt thống kê với 2 giá trị rh 50 và 51 với độ tin cậy 0.05- Tức là độ ẩm có ảnh hưởng đến nhiệt độ
```

Vậy ta có thể cho rằng độ ẩm có ảnh hưởng lên nhiệt độ, tức là phân phối hai giá trị độ ẩm 50 và 51 khác nhau đối với temp.

6. Trực quan hóa dữ liệu về hướng gió



Hình 19: Trực quan hướng gió

Hướng gió (wind_dir) được biểu diễn theo giá trị nguyên, rất khó hiểu nếu không xử lý thành hướng cụ thể. Bằng việc xử lý đặc trưng và trực quan hóa: Hướng gió chủ yếu ở Hoàn Kiếm là Đông Nam. Tốc độ gió chủ yếu từ 4-5 (đơn vị)

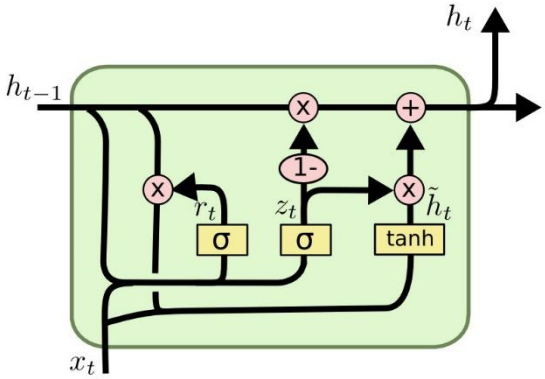
CHƯƠNG 6: XÂY DỰNG MÔ HÌNH

6.1. Mô hình xử lý bài toán Timeseries

Nhóm em quyết định sử dụng mô hình LSTM vì những lí do sau:

1. Mô hình Long Short-Term Memory (LSTM) là một dạng mạng nơ-ron hồi quy (RNN) được thiết kế đặc biệt để giải quyết vấn đề vanishing gradient trong quá trình huấn luyện mô hình. Mục tiêu chính của LSTM là nâng cao khả năng mô hình hóa chuỗi dữ liệu dài hạn bằng cách duy trì một trạng thái ẩn linh hoạt, giúp mô hình ghi nhớ thông tin quan trọng trong quá khứ và tích hợp nó vào các dự đoán tương lai.
2. Cấu trúc chính của LSTM bao gồm các cổng quan trọng, như cổng quên (forget gate), cổng đầu vào (input gate) và cổng đầu ra (output gate). Những cổng này cung cấp khả năng cho mô hình quyết định thông tin nào sẽ được giữ lại, thông tin nào sẽ bị loại bỏ, và thông tin nào sẽ được tích hợp vào trạng thái ẩn. Tính linh hoạt của cấu trúc này giúp LSTM đạt được hiệu suất cao khi xử lý các chuỗi dữ liệu có độ dài lớn và chứa các mối quan hệ lâu dài.

3. Mô hình LSTM có nhiều ứng dụng trong việc dự đoán chuỗi thời gian, xử lý ngôn ngữ tự nhiên và nhiều lĩnh vực khác. Trong dự đoán chuỗi thời gian, LSTM thể hiện khả năng mô hình hóa mối quan hệ phức tạp trong dữ liệu, trở thành một công cụ quan trọng trong việc dự đoán xu hướng và biểu đồ.
4. So với mô hình RNN truyền thống, LSTM có những ưu điểm nổi bật. Trong đó, khả năng xử lý các chuỗi dữ liệu dài hạn mà không gặp vấn đề vanishing gradient giúp LSTM duy trì khả năng học tốt trên các chuỗi dữ liệu lớn. Hơn nữa, cấu trúc ô cổng cho phép LSTM chủ động kiểm soát thông tin, giúp mô hình học được các mối quan hệ quan trọng và loại bỏ thông tin không cần thiết.



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$
$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$
$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Sơ đồ cấu trúc mô hình LSTM

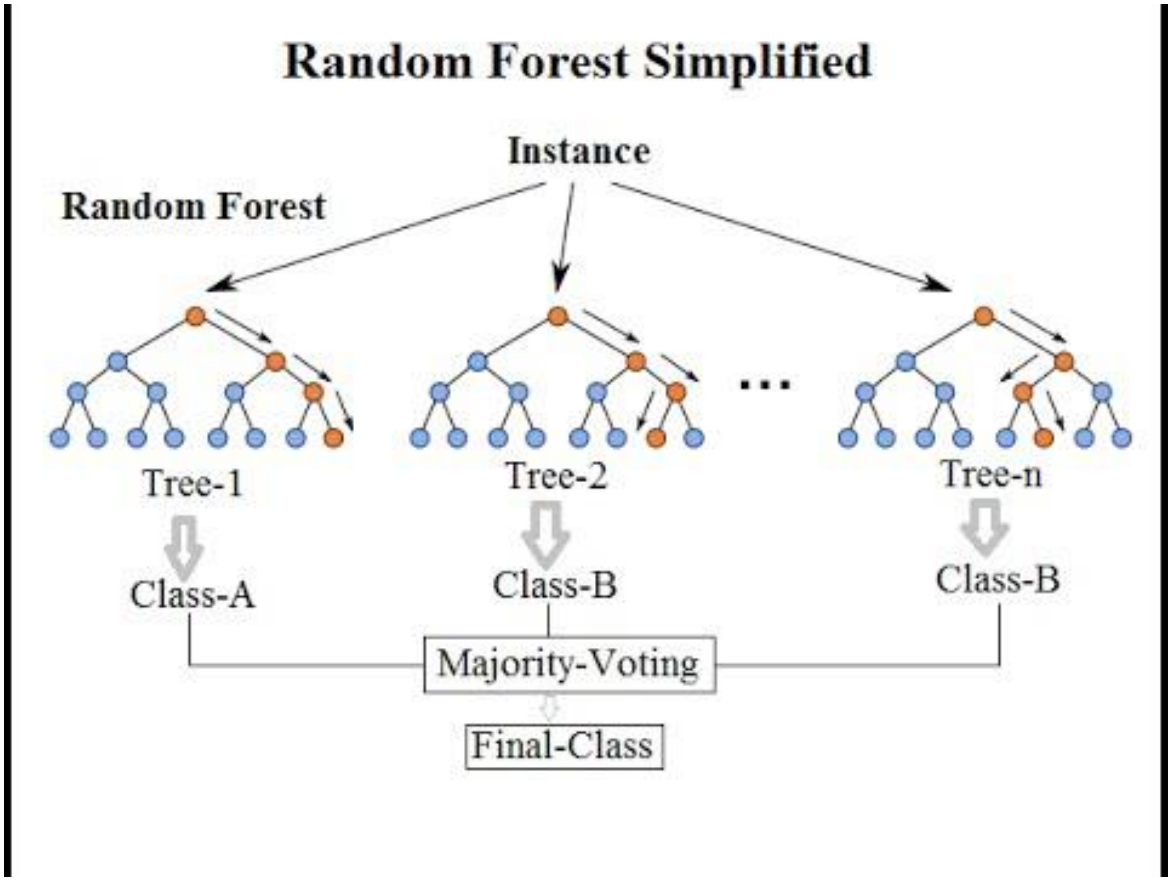
Kết quả thu được sau khi huấn luyện mô hình như sau:

6.2. Mô hình phân loại nhãn thời tiết

Nhóm em quyết định sử dụng mô hình Random forest classifier vì những lí do sau:

1. Random Forest là một tập hợp mô hình (ensemble). Mô hình Random Forest rất hiệu quả cho các bài toán phân loại vì nó huy động cùng lúc hàng trăm mô hình nhỏ hơn bên trong với quy luật khác nhau để đưa ra quyết định cuối cùng. Mỗi mô hình con có thể mạnh yếu khác nhau, nhưng theo nguyên tắc « wisdom of the crowd », ta sẽ có cơ hội phân loại chính xác hơn so với khi sử dụng bất kì một mô hình đơn lẻ nào.
2. Đơn vị của RF là thuật toán cây quyết định, với số lượng hàng trăm. Mỗi cây quyết định được tạo ra một cách ngẫu nhiên từ việc : Tái chọn mẫu (bootstrap, random sampling) và chỉ dùng một phần nhỏ tập biến ngẫu nhiên (random features) từ toàn bộ các biến trong dữ liệu. Ở trạng thái

sau cùng, mô hình RF thường hoạt động rất chính xác, nhưng đổi lại, ta không thể nào hiểu được cơ chế hoạt động bên trong mô hình vì cấu trúc quá phức tạp. RF do đó là một trong số những mô hình hộp đen (black box).



Sơ đồ cấu trúc mô hình Random Forest

Kết quả thu được sau khi huấn luyện mô hình như sau:

```
# Xây dựng mô hình Random Forest
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(x_train_normalized, y_train)

# Đánh giá mô hình trên tập kiểm tra
accuracy = model.score(x_test_normalized, y_test)S
print("Độ chính xác trên tập kiểm tra:", accuracy)
```

```
Độ chính xác trên tập kiểm tra: 0.9103738690035851
```

6.3. Tối ưu kiến trúc mạng LSTM

- Kiểm tra giá trị loss với số tầng LSTM:

Số tầng (hidden units 64)	Loss	MAE	val_loss	val_MAE
2	0.1212	0.2046	0.3154	0.3206
3	0.0537	0.1200	0.2826	0.2526
4	0.0425	0.1043	0.2954	0.2628
5	**0.0386**	**0.0971**	**0.2918**	**0.2660**

- Chọn số tầng là 5, kiểm tra giá trị loss với số hidden units:

Số hidden units	Loss	MAE	val_loss	val_MAE
16	0.1627	0.2082	0.2021	0.2255
32	**0.0931**	**0.1596**	**0.2644**	**0.2545**
64	0.0386	0.0971	0.2918	0.2660
128	0.0173	0.0637	0.2748	0.2468

- Chọn số tầng là 5, số hidden units là 32, kiểm tra giá trị loss với các giá trị dropout ở mỗi tầng:

Dropout	Loss	MAE	val_loss	val_MAE
20%	**0.1451**	**0.1966**	**0.2404**	**0.2384**
30%	0.1567	0.2057	0.2447	0.2460
40%	0.1669	0.2143	0.2475	0.2522
128	0.0173	0.0637	0.2748	0.2468

Bài toán sử dụng kiến trúc tối ưu là 5 tầng LSTM, 32 hidden units và kết hợp dropout 20% ở mỗi tầng.

6.4. Đánh giá mô hình RandomForest qua việc lựa chọn đặc trưng quan trọng

Loại bỏ đặc trưng	**Kết quả trên tập test**
Clouds	0.781
Precip	0.911
solar_rad	0.996

Như vậy, các đặc trưng clouds và precip là quan trọng nhất để dự đoán nhãn thời tiết.

CHƯƠNG 7: XÂY DỰNG HỆ THỐNG

Nhóm bọn em lên kế hoạch sẽ sử dụng Airflow để lên lịch lấy dữ liệu và train lại model trên dữ liệu mới.

Apache Airflow là một công cụ mã nguồn mở được sử dụng để lên lịch và tự động hóa các quy trình làm việc. Nó giúp quản lý, theo dõi và lên lịch các công việc có thể thực hiện tự động trong hệ thống thông tin.

- Apache Airflow là một công cụ mã nguồn mở được sử dụng để **lập lịch, quản lý, và giám sát các quy trình xử lý dữ liệu**. Nó được sử dụng rộng rãi trong các hệ thống xử lý dữ liệu lớn để tự động hóa các quy trình xử lý dữ liệu phức tạp.
- Airflow cung cấp các khái niệm như "DAG" (Directed Acyclic Graph), "Task", "Operator", "Sensor" để mô tả các quy trình xử lý dữ liệu.
- Xử lý lỗi và tái chạy: Nếu một công việc thất bại, Airflow cung cấp khả năng xử lý lỗi và tái chạy các công việc từ giai đoạn thất bại.
- Quản lý kết quả và giữ trạng thái: Airflow theo dõi trạng thái của các công việc và cung cấp giao diện để xem kết quả và tiến độ của chúng.
- Mở rộng và tùy chỉnh: Airflow có thể được mở rộng thông qua việc sử dụng nhiều tiện ích và tính năng mà cộng đồng người dùng đóng góp. Nó cũng hỗ trợ việc tùy chỉnh để đáp ứng nhu cầu cụ thể của các tổ chức.
- Giao diện web: Airflow cung cấp giao diện web để theo dõi và quản lý các DAG và công việc.

Nhóm có kế hoạch lên lịch crawl 1 ngày 1 lần và sẽ huấn luyện lại mô hình sau 1 tuần

CHƯƠNG 8: Kết luận

8.1 Các khó khăn và hướng phát triển trong tương lai

Do nhóm sử dụng dịch vụ miễn phí nên số lượng data có thể crawl bị hạn chế.

Tập dữ liệu có độ mất cân bằng cao, cần phải được xử lý kỹ càng để cho vào mô hình để huấn luyện

8.2 Kết luận

Trong báo cáo này, nhóm chúng em đã nghiên cứu về dự đoán thời tiết tại Hà Nội bằng cách áp dụng khoa học dữ liệu. Nhóm đã thu thập và phân tích một tập dữ liệu lớn về các biến số thời tiết, như nhiệt độ, độ ẩm, áp suất không khí và tốc độ gió, từ các trạm quan trắc khí tượng trong một khoảng thời gian dài. Sử dụng các kỹ thuật phân tích dữ liệu và mô hình hóa, Nhóm đã xây dựng một mô hình dự đoán thời tiết cho Hà Nội.