МУИС - ХШУИС - Компьютерийн сүлжээ
Нэр : А. Мэндбаяр
ID : 19b1num0239

Дохио ба систем - Мэргэжлийн удиртгал

Цаг уурын алдаатай хэмжигдэхүүнийг
машин сургалтын аргаар тодорхойлох

2022 он

Агуулга:

Outlier гэж юу вэ?

“Data Analyst болон Data Scientist – ууд нийт ажлынхаа 60 орчим хувийг датагаа цэвэрлэх, хувиргах болон боловсруулах ажилд зарцуулдаг”. Outlier нь их хэмжээний өгөгдөлтэй ажиллах үед таарч болох гажууд өгөгдлүүдийг хэлэх ба бусад утгуудаасаа хэт хол байх ба энэ нь машин сургалт болон шинжилгээнд ашиглах үед загварын дунджыг хиймлээр өсгөж гажуудуулах нөхцөлийг бий болгодог. Тиймээс outlier утгыг цэвэрлэх нь маш том ач холбогдолтой юм.

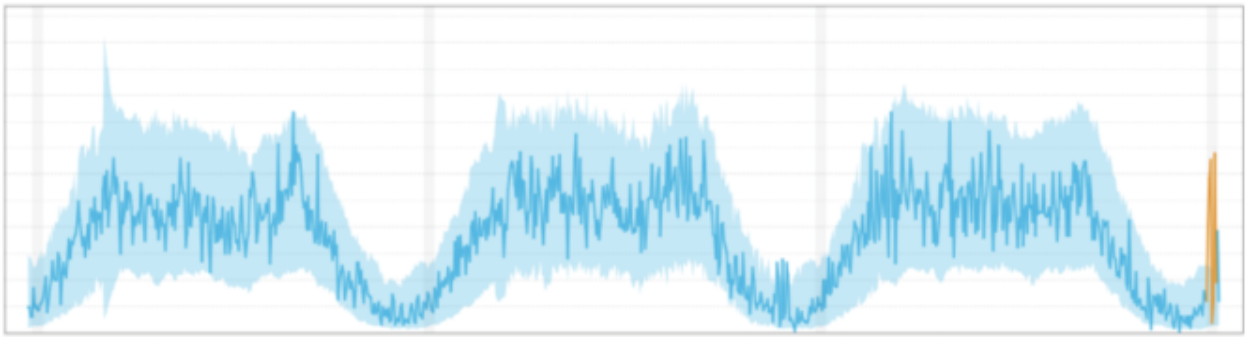
Outlier - ийн төрлүүд

1. **Global outlier:**

   Хэрэв өгөгдлийн утгын цэг нь тухайн өгөгдлийн багцын утгаас хол давсан бол өгөгдлийн цэгийг Global outlier гэж үзнэ.

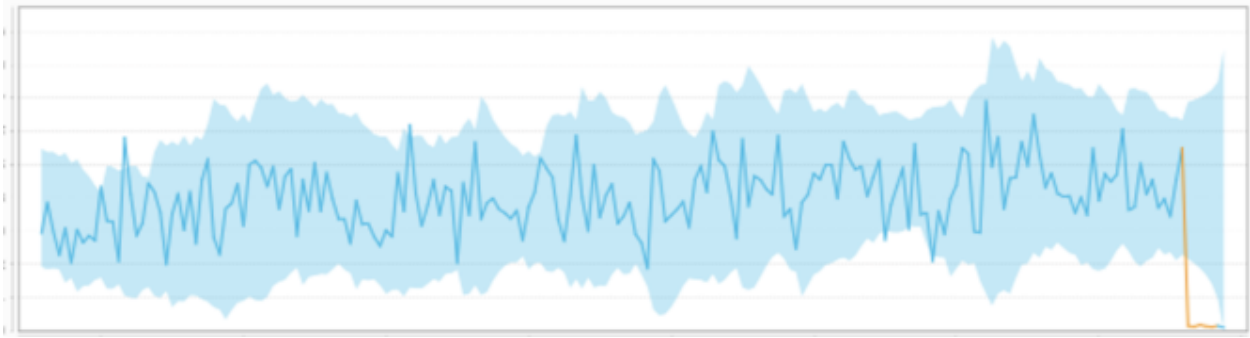
2. **Contextual Outliers:**

   Өгөгдлийн утгуудын цэгүүд нь ижил төстэй бусад өгөгдлөөс ихээхэн зөрүүтэй байхыг хэлнэ. Ихэвчлэн цаг хугацаагаанаас хамаарсан өгөгдөлд илэрдэг outliers.



3. **Collective Outliers:**

Хэсэг бүлэг багц өгөгдөл бусад өгөгдлүүдийн утгаас мэдэгдэхүйц зөрсөн байхыг Collective outliers гэнэ. Гэхдээ бие даасан өгөгдлийн утгын цэгүүд нь Contextual болон Global outlier - д тооцогддоггүй.



Ашиглагдах өгөгдөл

Ашиглагдах өгөгдлүүд нь Температур, Чийгшил болон Даралт гэсэн 3 утга байх ба энэ нь тус бүр 192718 мөр урт юм.

```
humid = pd.read_csv('/home/barkowich/Documents/anaconda/ml/weather/21h.csv')
press = pd.read_csv('/home/barkowich/Documents/anaconda/ml/weather/21p.csv')
tempr = pd.read_csv('/home/barkowich/Documents/anaconda/ml/weather/21t.csv')
```

```
merged = humid
merged = merged.join(press, lsuffix="_left")          #мус бүрийн багануудыг нэгтгэнэ
merged = merged.join(tempr, lsuffix="_left" )
merged.shape                                          #Өгөгдлийн хэлбэр хэмжээг үзүүлнэ
```

```
(192718, 3)
```

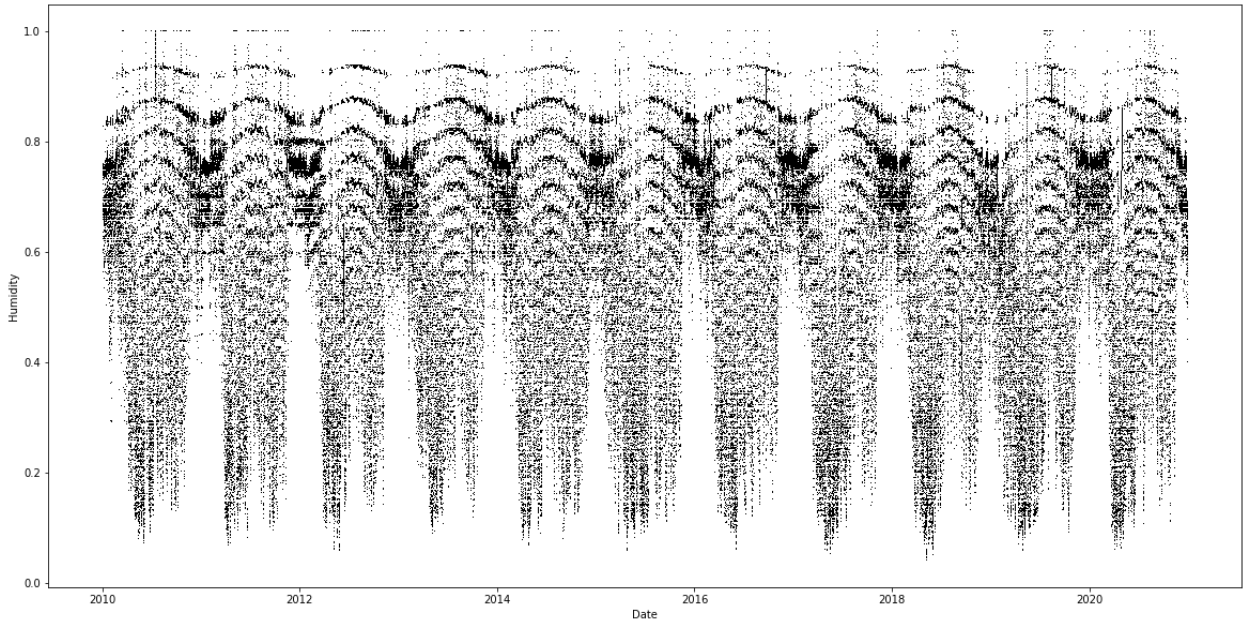
Зураг 1 - цаг уурын датаг оруулж тус бүрийг нэгтгэж нэг болгож буй код

Эдгээр өгөгдлүүд нь Улаанбаатар дахь цаг уурын газарын 2010 - 2020 он хүртэлх 30 минут тутамд хэмжиж авч цуглуулсан өгөгдөл ба тухайн ***merged*** DataFrame - д цаг хугацааны мэдээллийг оруулж өгснөөр ***Plot*** хийж өгөгдлийн утгыг дүрслэх боломжтой болно.

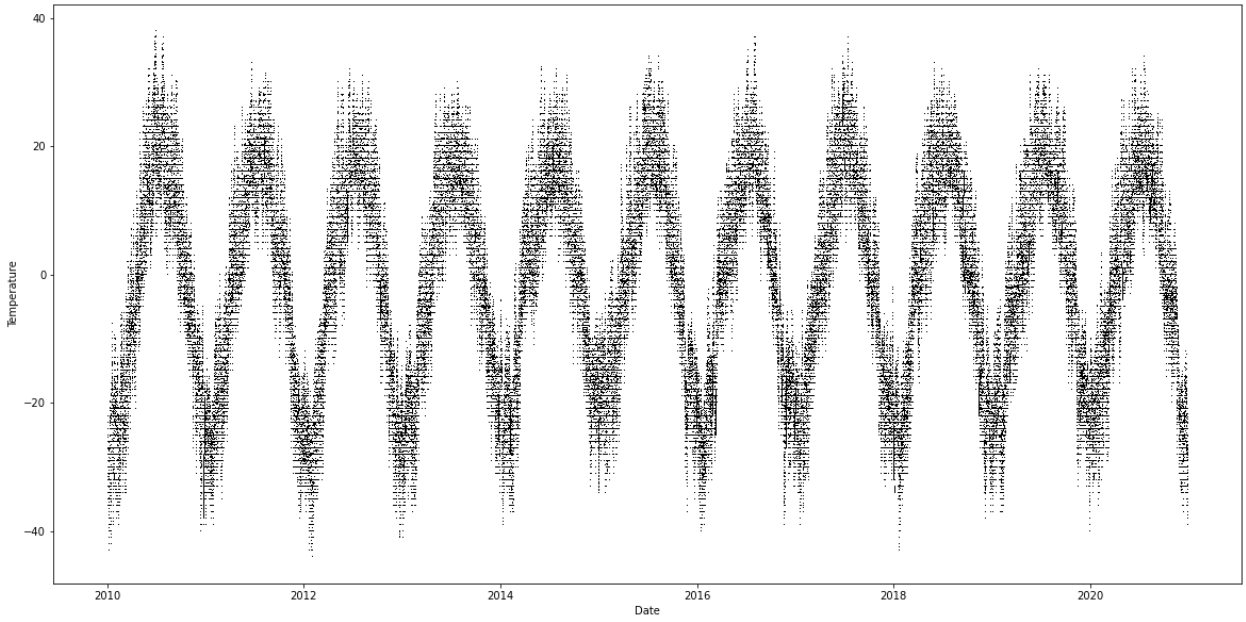
```
df = pd.DataFrame()
df['Timestamp'] = pd.DataFrame(pd.date_range(start="2010-01-01", end="2020-12-31", freq='30T'))
merged = merged.join(df, how = 'left')
merged.head()
```

	Humidity	Pressure	Temperature	Timestamp
0	0.742	1037.400000	-33.0	2010-01-01 00:00:00
1	0.742	1037.033333	-33.0	2010-01-01 00:30:00
2	0.668	1036.666667	-34.0	2010-01-01 01:00:00
3	0.742	1036.300000	-33.0	2010-01-01 01:30:00
4	0.673	1035.933333	-32.0	2010-01-01 02:00:00

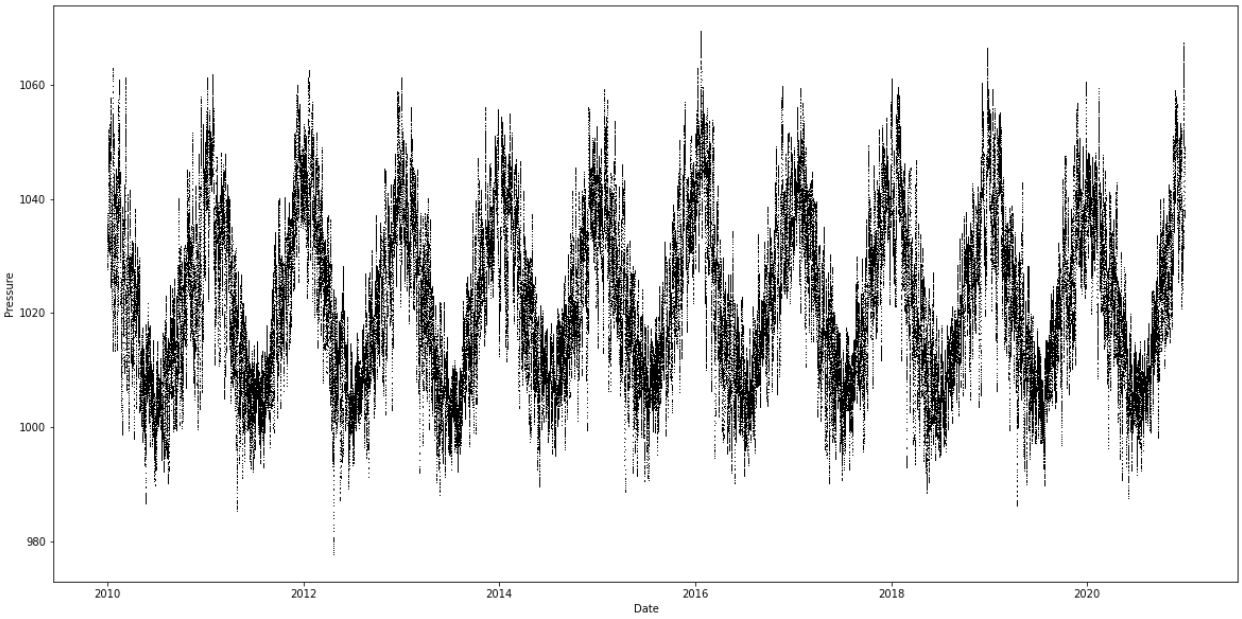
Зураг 2 - merged DataFrame - д цаг хугацааны багана нэмж оруулах



Зураг 3 - Чийгшил (Humidity) - өгөгдлийг цаг хугацаатай нь харгалзуулан гаргасан зураг



Зураг 4 - Температур (Temperature) - өгөгдлийг цаг хугацаатай нь харгалзуулан гаргасан зураг

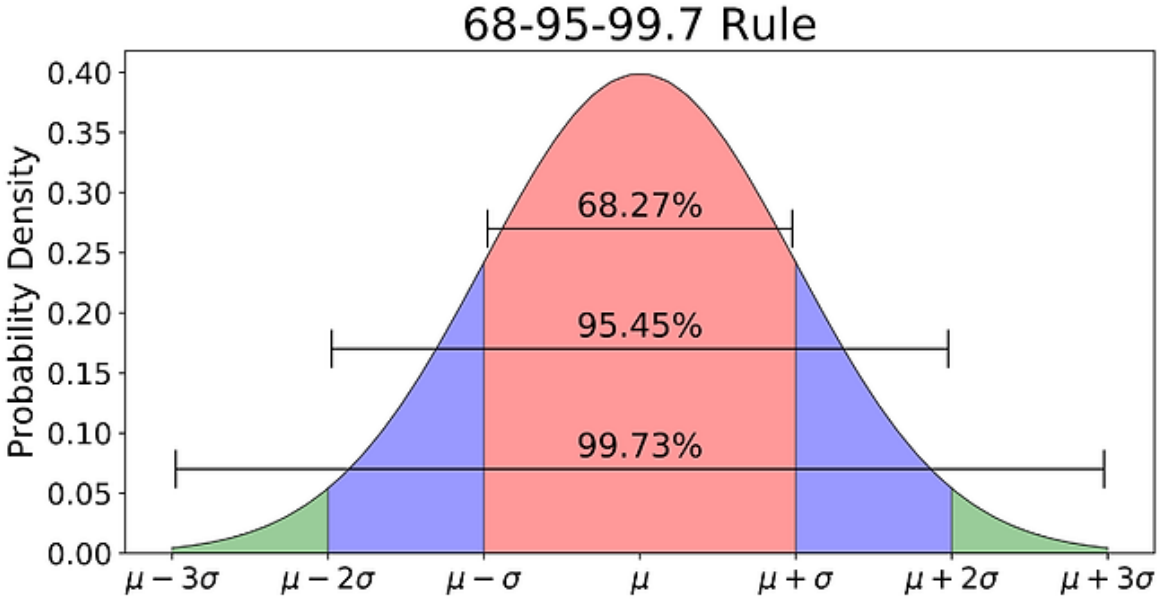


Зураг 5 - Даралт (Pressure) - өгөгдлийг цаг хугацаатай нь харгалзуулан гаргасан зураг

Outlier detect:

Merged буюу 3 багана өгөгдлийг нэгтгэж цаг хугацааны багана нэмж өгсөн DataFrame нь өөртөө бага хэмжээний outlier - ыг агуулна. Outlier - н төрлийг Global outlier гэж үзсэн ба үүнийг standard deviation болон Z_score аргаар олох боломжтой.

Standard Deviation: Стандарт хазайлт буюу стандарт хазайлт гэсэн нэр томъёо нь тоон өгөгдлийн хэлбэлзэл эсвэл тархалтыг хэмжихэд ашигладаг хэмжүүрийг хэлнэ. Эхлээд багана бүрийн дундаж утгыг тооцоолж, дараа нь дисперсийг, эцэст нь стандарт хазайлтыг тооцоолно. Тухайн гаргаж авсан стандарт хазайлт нь 3 - аас их болон -3 аас бага байгаа тохиолдолд outlier гэж үзнэ.



Зураг 10 - Стандарт хазайлтыг тооцоолох дүрэм. Нормал тархалттай хэмжигдэхүүний **99.7%** нь дунджаас доош болон дээш 3 стандарт хазайлттай тэнцэх зайд багтдаг.

```
# outlier detection
def standart_division(data, threshhold = 3):
 upper_limit = data.mean() + threshhold * data.std()
 lower_limit = data.mean() - threshhold * data.std()
 return data[(data>upper_limit) | (data<lower_limit)]
```

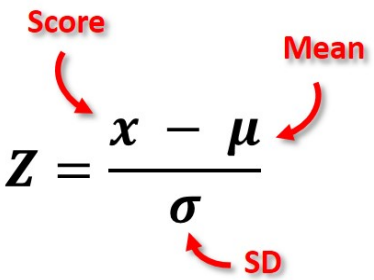
```
[141] standart_division(merged.Humidity, 3)

      Series([], Name: Humidity, dtype: float64)
```

```
[142] standart_division(merged.Pressure).head()

      40395      977.500
      106183    1066.700
      106184    1066.925
      106185    1067.150
      106186    1067.375
      Name: Pressure, dtype: float64
```

```
[143] standart_division(merged.Temperature)

      Series([], Name: Temperature, dtype: float64)
```

Зураг 11 - Стандарт хазайлтыг олох функцыг бичиж түүгээр багана тус бүрийн outlier - ийг тодорхойлсон

Z_score: Z-score буюу стандарт оноо нь өгөгдсөн өгөгдлийн утгын цэгийн дунджаас дээгүүр эсвэл доогуур байгаа стандарт хазайлтын тоо юм. Дундаж нь бүлгийн бүх утгуудын дунджийг нэгтгэж, дараа нь бүлгийн нийт зүйлийн тоонд хуваана. Тэрхүү гаргаж авсан дундаж утгыг тухайн өгөгдлөөс хасч гарсан хариуг стандарт хазайлтад нь хувааж өгснөөр Z_score нь гарах ба энэ нь -3<=x<= 3 утгуудаас гадна байгаа тохиолдолд outlier гэж үзнэ.



```python
# Z-score арааар outliers - ийг илрүүлэх

def z_score(data, name,threshhold=3):
  data_zscore = pd.DataFrame()
  data_zscore['Data'] = data
  data_zscore['zscore'] = ( data_zscore.Data - data_zscore.Data.mean() ) / data_zscore.Data.std()
  return data_zscore[(data_zscore.zscore < -threshhold) | (data_zscore.zscore > threshhold)]
```

```python
z_score(merged.Humidity, 'Humidity',3)
```

	Data	zscore

```python
raw_dat = z_score(merged.Pressure, 'Pressure')
# Зөвхөн даралтын утгаас outliers илэрсэн
```

```python
z_score(merged.Temperature, 'Temperature')
```

	Data	zscore

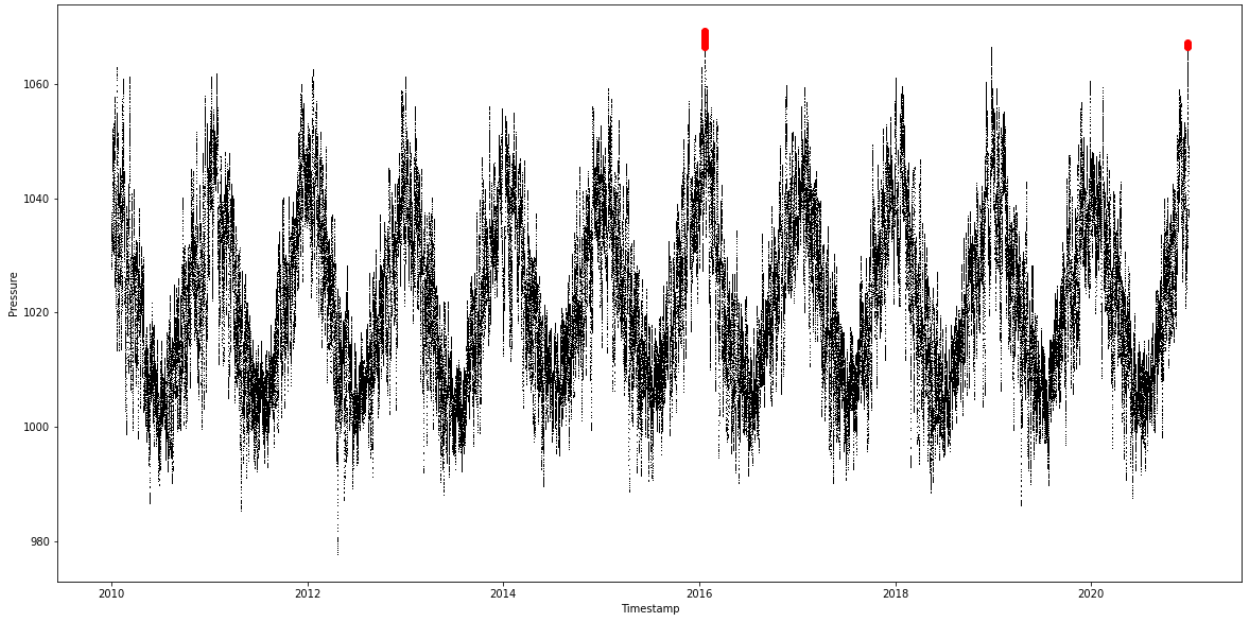
```python
raw_dat['outlier'] = 1
merged = merged.join(raw_dat.outlier).fillna(0)
merged[merged['outlier']==1].sum()
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:3: FutureWarning: Dropping of nuisance
  This is separate from the ipykernel package so we can avoid doing imports until
Humidity          30.235500
Pressure       51160.166667
Temperature    -1493.400000
outlier           48.000000
dtype: float64
```

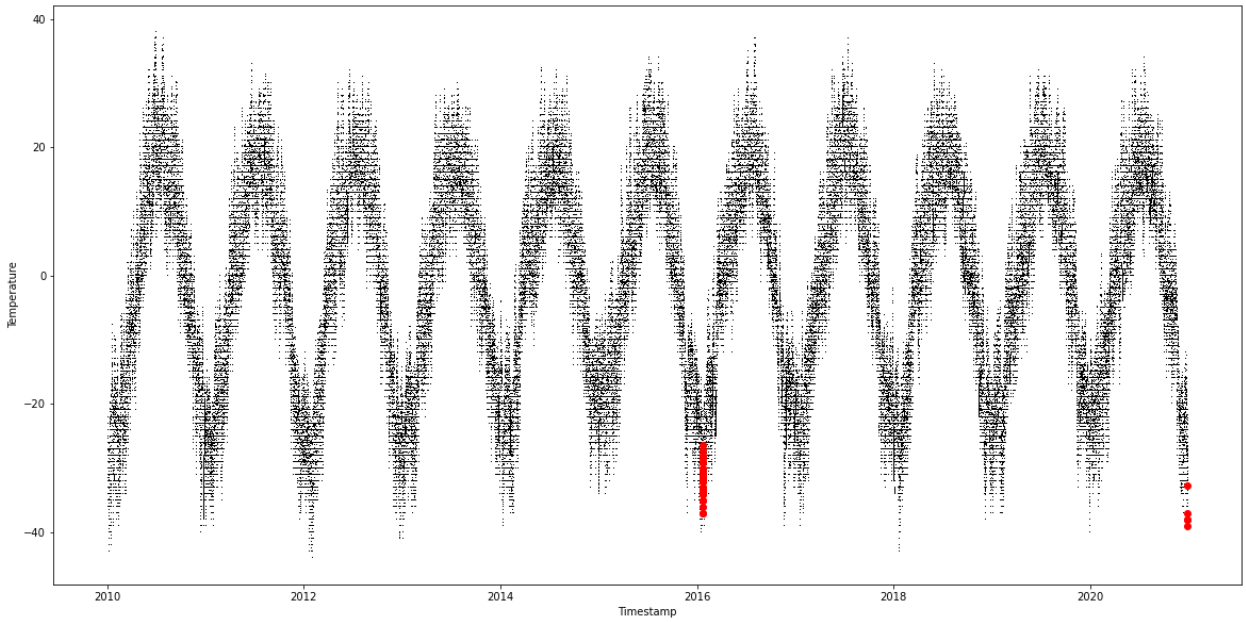
Зураг 12 - Z_score олох функц бичиж түүгээр багана тус бүрийн outlier - ыг шүүсэн

Багана тус бүрд Z_score аргыг хэрэгжүүлэх үед зөвхөн даралтын утгаас outlier илэрсэн ба үүнийг **raw_dat** DataFrame - ын outlier баганад оруулж өгсөн.

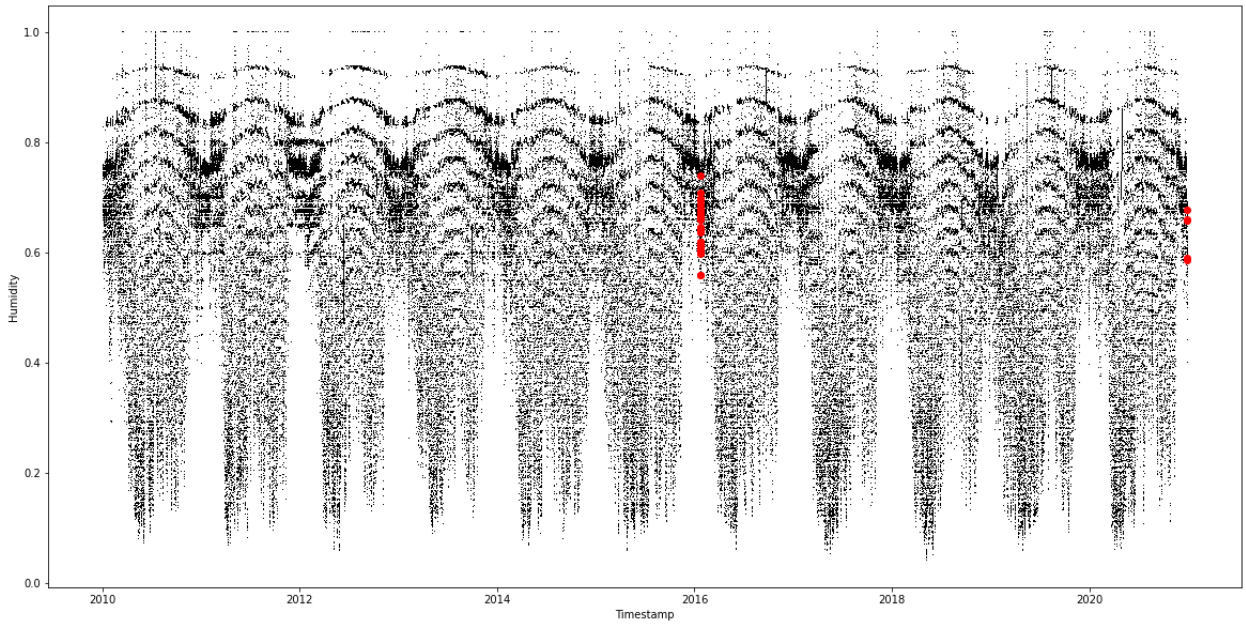
Илэрсэн Outlier: Нийт 48 outlier илэрсэн. Тухайн outlier нь даралтын утга (Pressure) - аас илэрсэн ба outlier гэж тооцогдсон утгыг тухайн мөртэй нь цуг устгах тул нэг цагт хэмжигдсэн температур болон чийгшилын утга мөн адил устана.



Зураг 13 - Даралтын утгаас илэрсэн outlier



Зураг 14 - Температурын утгаас илэрсэн outlier



Зураг 15 - Чийгшил - ын утгаас илэрсэн outlier

```
raw_dat['outlier'] = 1
merged = merged.join(raw_dat.outlier).fillna(0)
merged[merged['outlier']==1].sum()
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_laun
  This is separate from the ipykernel package so we c
Humidity          30.235500
Pressure       51160.166667
Temperature    -1493.400000
outlier           48.000000
dtype: float64
```

Зураг 16 - Илэрсэн Outlier - ыг 1 гэж үзэн энгийн өгөгдлийг 0 ээр илэрхийлэх outlier гэх баганыг үндсэн merged DataFrame рүү нэмнэ

```
[150] data
```

	Humidity	Temperature	Pressure	outlier
0	0.742	-33.0	1037.400000	0.0
1	0.742	-33.0	1037.033333	0.0
2	0.668	-34.0	1036.666667	0.0
3	0.742	-33.0	1036.300000	0.0
4	0.673	-32.0	1035.933333	0.0
...	...	...	...	...
192713	0.688	-26.0	1037.500000	0.0
192714	0.686	-27.0	1037.350000	0.0
192715	0.623	-27.0	1037.200000	0.0
192716	0.683	-28.0	1037.050000	0.0
192717	0.599	-23.2	1036.900000	0.0

192718 rows × 4 columns

Зураг 17 - Outlier баганыг үндсэн DataFrame рүү нэмж өгсөний дараах өгөгдөл

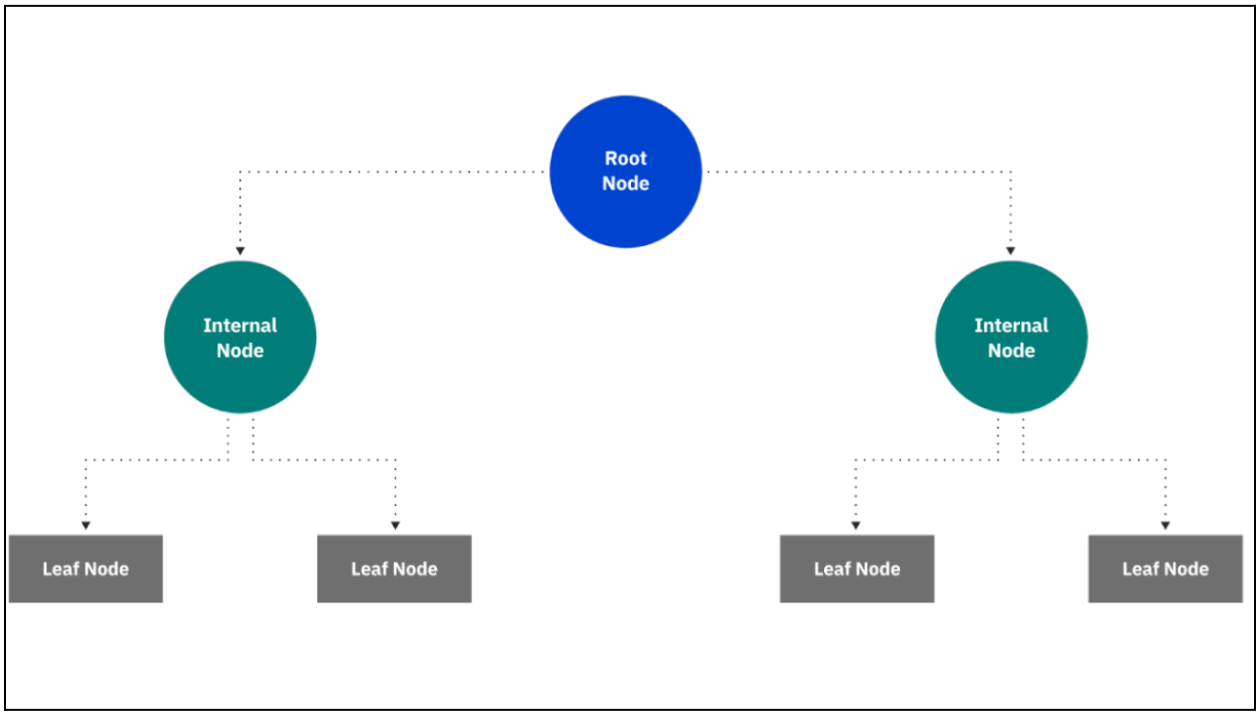
DecisionTree машин сургалт

Машин сургалт гэж юу вэ: Машин сургалт гэдэг нь компьютерыг өөрөө туршлагаасаа сурах боломжийг олгож буй аргачлал юм. Өөрөөр хэлбэл, компьютер өөрт өгөгдсөн асуудлыг шийдэхдээ өөрийн туршлага дээр үндэслэн өөрөө тухайн асуудлыг шийдэж сурах юм.

Машин сургалт нь дараах 4 төрөлтэй:

Зураг 18 - Машин сургалтын төрлүүд source - (https://medium.com/@minerva2991)

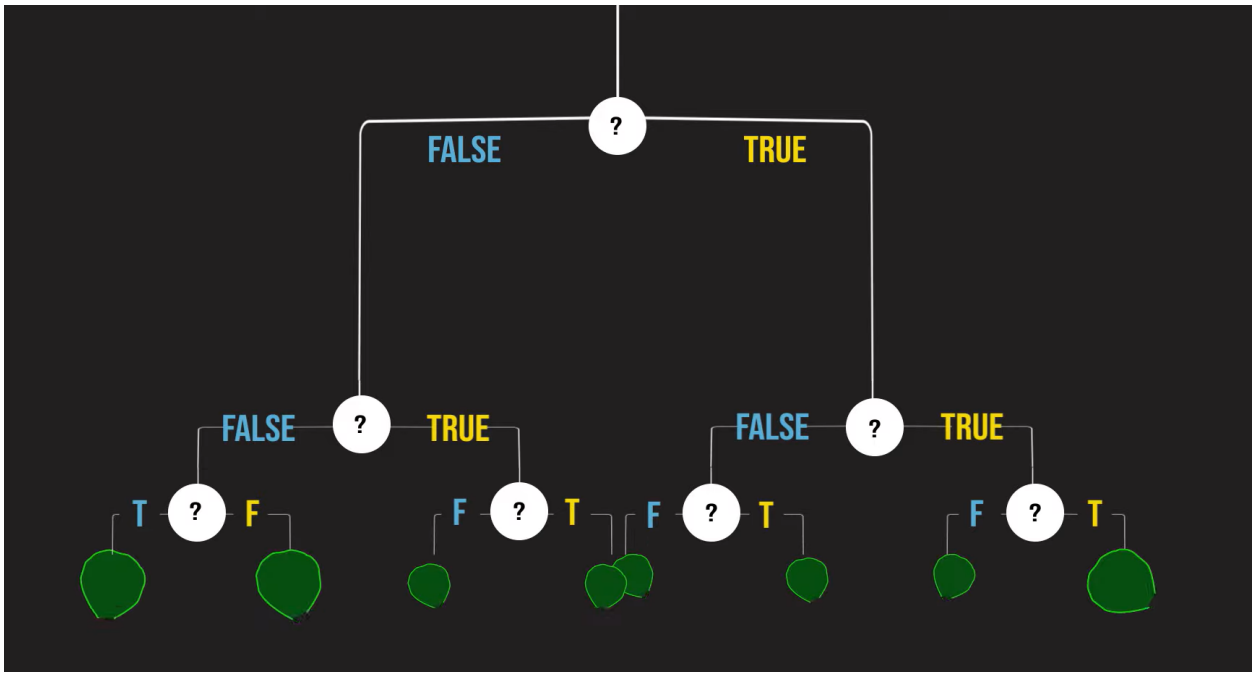
DecisionTree: DecisionTree нь ангилал болон регрессийн даалгаварт ашиглагддаг параметрийн бус Supervised сургалтын алгоритм юм. Үндэс(root node), мөчир(branch), дотоод зангилаа(nodes), навч(leaf) зэргээс бүрдэх шаталсан, модны бүтэцтэй.



Зураг 19 - DecisionTree бүтэц source - (https://www.ibm.com/topics/decision-trees)

Зураг 18 - д үзүүлсэн диаграммаас харахад DecisionTree нь root node - с эхэлдэг. Root node - с гарч буй салбарууд нь internal node руу дамждаг бөгөөд үүнийг шийдвэрийн зангилаа гэж нэрлэдэг. Боломжит боломжууд дээр үндэслэн зангилааны төрөл хоёулаа навчны зангилаа эсвэл төгсгөлийн зангилаагаар тэмдэглэгдсэн нэгэн төрлийн дэд олонлогуудыг үүсгэхийн тулд үнэлгээ хийдэг. Навч зангилаа нь өгөгдлийн багц доторх бүх боломжит үр дүнг илэрхийлдэг. Outlier илрүүлэх датаны хувьд 2 төрлийн боломж байгаа нь 0 болон 1 юм.

DecisionTree нь Root node - с эхлүүлэн бүх label - уудыг оролцуулан гаралт нь True болон False гарах асуултыг node бүр дээр асууж цааш дамжуулна.



Зураг 20 - DecisionTree

```
[ ]  from sklearn.model_selection import train_test_split
     from sklearn.tree import DecisionTreeClassifier
     from sklearn.metrics import accuracy_score
     from sklearn import tree

[ ]  X = data.iloc[:, :-1].values
     Y = data.iloc[:, -1].values
     X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=.3, random_state=41)

[ ]  clf_entropy = DecisionTreeClassifier(criterion = "entropy", random_state = 100,
     max_depth=3, min_samples_leaf=5)
     clf_entropy.fit(X_train, Y_train)

     DecisionTreeClassifier(criterion='entropy', max_depth=3, min_samples_leaf=5,
                            random_state=100)

[ ]  y_pred_en = clf_entropy.predict(X_test)
     y_pred_en

     array([0., 0., 0., ..., 0., 0., 0.])

[ ]  accuracy_score(Y_test, y_pred_en)

     0.9999654074996541
```

Зураг 21 - DecisionTree машин сургалтыг сургах код

Зураг 21 - т DecisionTree машин сургалтын аргыг хэрэглэх үед шаардлагатай sklearn сангийн функцуудыг татаж авч оруулан ажиллуулсан ба үүнд:

- train_test_split : өгөгдлийг сургах болон турших гэсэн 2 хэсэгт хуваах
- DecisionTreeClassifier : DecisionTree алгоритм
- accuracy_score : сургалтын чанарыг шалгах алдааны хувийг харах

гэсэн функцууд байна.

```
[ ]  X = data.iloc[:, :-1].values
     Y = data.iloc[:, -1].values
     X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=.3, random_state=41)
```

Зураг 22 - Өгөгдлийг хуваах

Дээрх зурагт машиныг сургах датаг сургалт болон шалгах гэсэн 2 хэсэгт хуваасан ба шалгах буюу test_size нь нийт датаны 30% байна. X - нь outlier баганаас бусад багана ба Y - нь зөвхөн outlier баганы утгуудыг хадгална.

```
[ ]  clf_entropy = DecisionTreeClassifier(criterion = "entropy", random_state = 100,
     max_depth=3, min_samples_leaf=5)
     clf_entropy.fit(X_train, Y_train)

     DecisionTreeClassifier(criterion='entropy', max_depth=3, min_samples_leaf=5,
                            random_state=100)
```

Зураг 23 - DecisionTree алгоритмыг ашиглаж машин сургалт ажиллуулах

Дээрх зурагт үзүүлснээр clf_entropy нэрээр зарлагдсан функцэд DecisionTree алгоритмыг өгч хадгална. Тэгэхдээ

- Criterion = ‘entropy’ : Тодорхойгүй байдлыг хэмждэг мэдээллийн онолын хэмжүүр. Энэ нь DecisionTree өгөгдөл хуваахыг хэрхэн сонгохыг тодорхойлдог.
- Random_state = ‘100’ : Дурын тоо
- max_depth = 3 : Тухайн модны салаалж буух мөчрийн тоо
- min_samples_leaf = 5: навчны зангилаанд байх шаардлагатай сорилын хамгийн бага тоог заана

```
[ ]  y_pred_en = clf_entropy.predict(X_test)
     y_pred_en

     array([0., 0., 0., ..., 0., 0., 0.])

[ ]  accuracy_score(Y_test, y_pred_en)

     0.9999654074996541
```

Зураг 24 - Машин сургалтын сургасны дараах accuracy

Дээрх зурагт үзүүлснээр DecisionTree - г ашиглан X_test - ыг шалгасан ба шалгасан үзүүлэлтийг y_pred_en - д хадгалж accuracy_score - оор Y_test - ын утгатай харьцуулж үзэх үед 99.99% - ийн харьцаатай гэж гарсан.

```
[36] all_ds = pd.DataFrame()
     all_in_one = data.iloc[:, :-1].values


     dataset = data.iloc[:, :-1]
     dataset['outlier'] =  clf_entropy.predict(all_in_one)
```

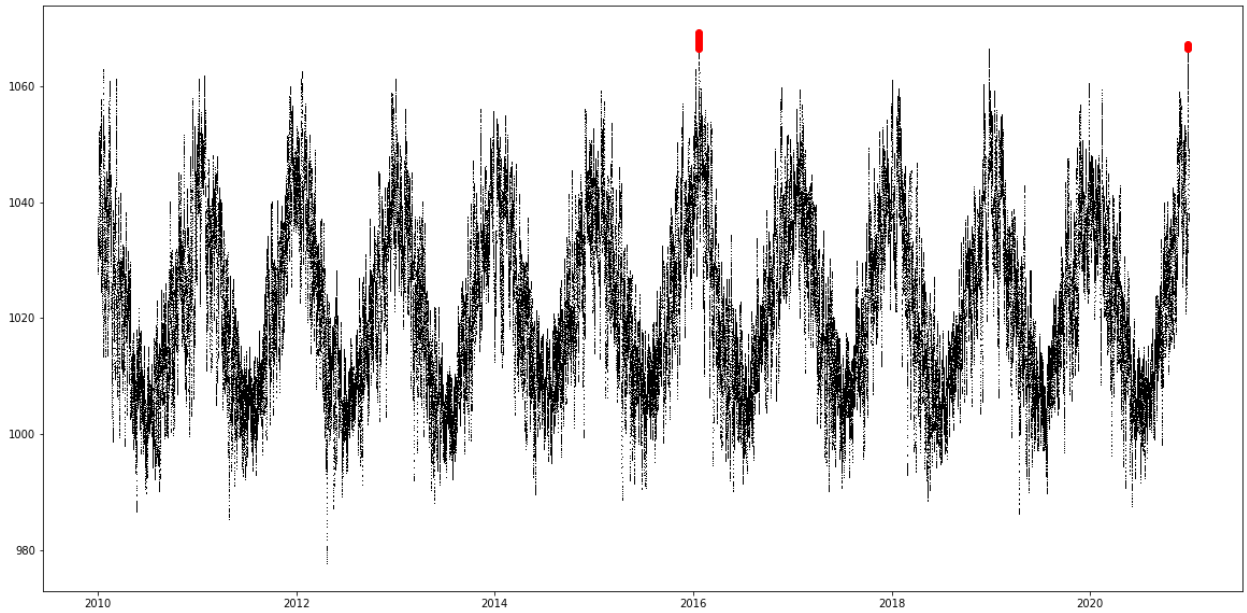
```
[37] dataset.outlier.sum()

     48.0
```

Зураг 25 - үндсэн датаг машин сургалтаараа тестлех

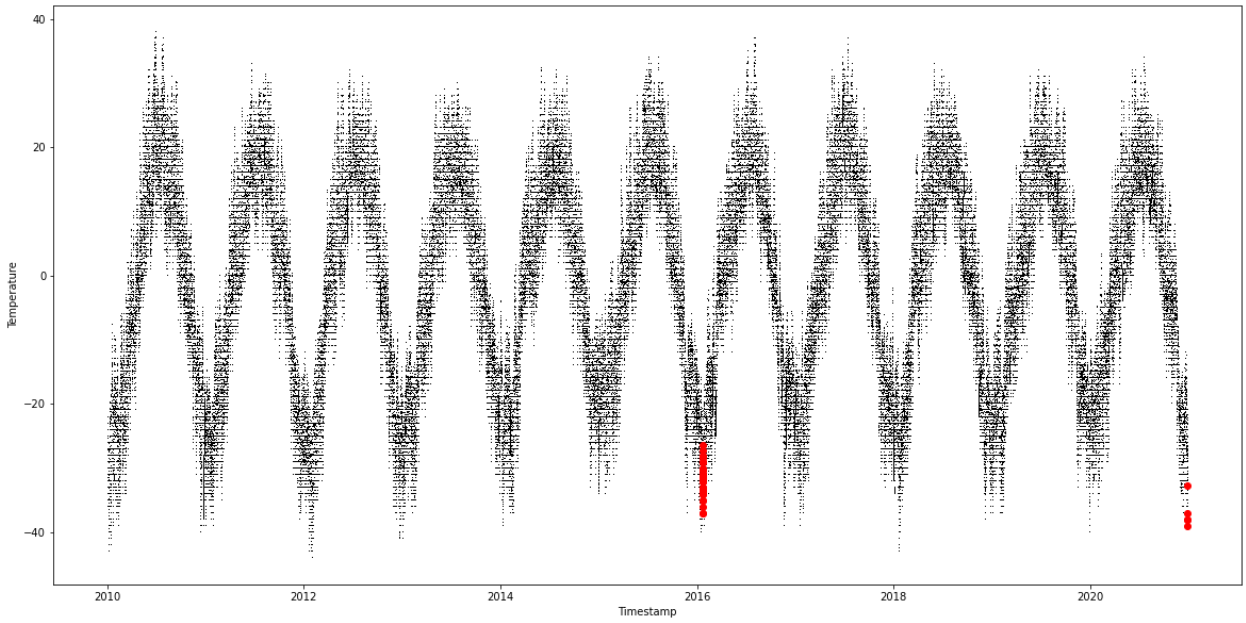
Дээрх зурагт үзүүлснээр **all_in_one** буюу үндсэн 3 багана өгөгдлийг машин сургалтруу оруулж outlier - ыг тодорхойлох гэж оролдов. Таасан outlier - ын утгуудыг **dataset** DataFrame - ийн outlier гэсэн баганад хадгалав.

Үр дүн:

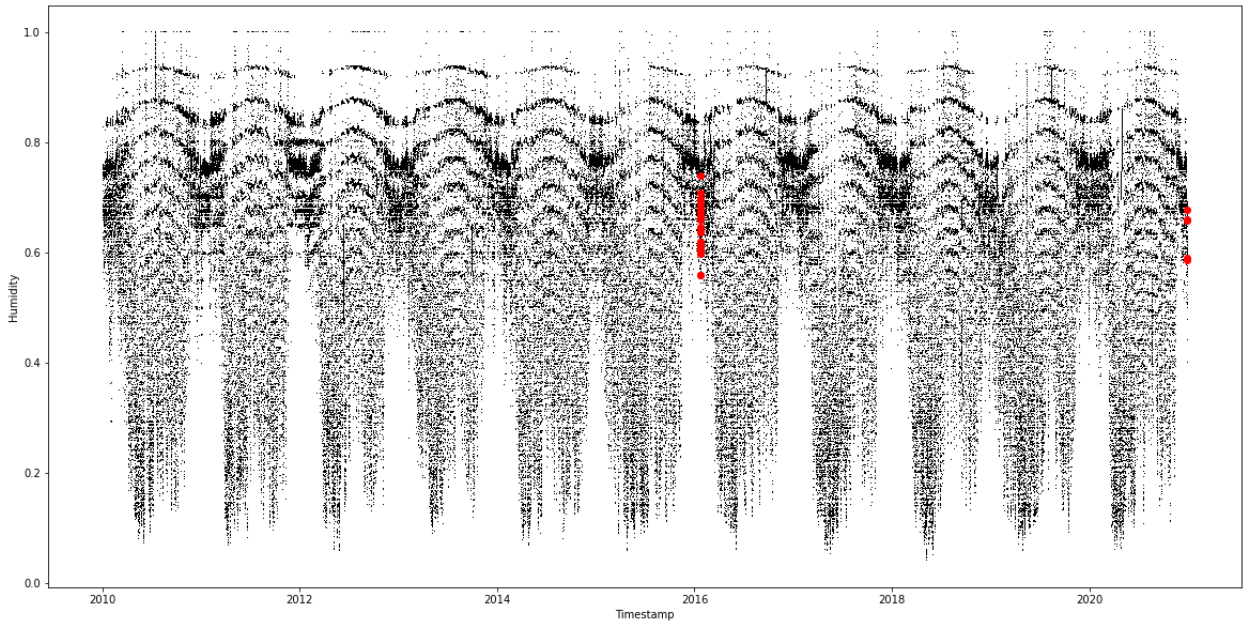
Машин сургалтын аргаар outlier утгуудыг олсныг доорх зургуудад дүрслэв.



Зураг 26 - Машин сургалтын аргаар тодорхойлсон даралтын outlier



Зураг 27 - Машин сургалтын аргаар тодорхойлсон температурын outlier



Зураг 28 - Машин сургалтын аргаар тодорхойлсон чийгшилын outlier

```
detected
```

	data	z_score	real_outlier	pred_outlier
40395	977.500000	-3.018437	1	0.0
106183	1066.700000	3.011648	1	1.0
106184	1066.925000	3.026859	1	1.0
106185	1067.150000	3.042069	1	1.0
106186	1067.375000	3.057279	1	1.0
106187	1067.600000	3.072490	1	1.0
106188	1067.825000	3.087700	1	1.0
106189	1068.050000	3.102911	1	1.0
106190	1068.275000	3.118121	1	1.0
106191	1068.500000	3.133332	1	1.0
106192	1068.233333	3.115304	1	1.0
106193	1067.966667	3.097277	1	1.0
106194	1067.700000	3.079250	1	1.0
106195	1067.433333	3.061223	1	1.0
106196	1067.166667	3.043196	1	1.0
106197	1066.900000	3.025169	1	1.0
106213	1066.833333	3.020662	1	1.0
106214	1067.516667	3.066856	1	1.0
106215	1068.200000	3.113051	1	1.0
106216	1068.383333	3.125445	1	1.0
106217	1068.566667	3.137838	1	1.0
106218	1068.750000	3.150232	1	1.0
106219	1068.933333	3.162626	1	1.0
106220	1069.116667	3.175019	1	1.0
106221	1069.300000	3.187413	1	1.0
106222	1069.116667	3.175019	1	1.0
106223	1068.933333	3.162626	1	1.0
106224	1068.750000	3.150232	1	1.0
106225	1068.566667	3.137838	1	1.0
106226	1068.383333	3.125445	1	1.0
106227	1068.200000	3.113051	1	1.0
106228	1068.033333	3.101784	1	1.0
106229	1067.866667	3.090517	1	1.0
106230	1067.700000	3.079250	1	1.0
106231	1067.533333	3.067983	1	1.0
106232	1067.366667	3.056716	1	1.0
106233	1067.200000	3.045449	1	1.0
106234	1067.116667	3.039816	1	1.0
106235	1067.033333	3.034182	1	1.0
106236	1066.950000	3.028549	1	1.0
106237	1066.866667	3.022915	1	1.0
106238	1066.783333	3.017282	1	1.0
106239	1066.700000	3.011648	1	1.0
192588	1066.550000	3.001508	1	1.0
192589	1066.766667	3.016155	1	1.0
192590	1066.983333	3.030802	1	1.0
192591	1067.200000	3.045449	1	1.0
192592	1066.666667	3.009395	1	1.0

Зураг 29 - Бодит Outlier ба машин сургалтын аргаар тодорхойлсон outlier - ыг харьцуулсан DataFrame

Дүгнэлт:

Энэхүү машин сургалтын аргаар цаг агаарын алдаатай хэмжигдэхүүнийг олох project нь Supervised машин сургалтын аргаар явагдав. Outlier - г олохдоо standard deviation болон z_score аргачлалыг ашиглаж Humidity, Temperature болон Pressure гэсэн 3 хэмжигдэхүүн тус бүрээс шүүн илрүүлсэн. Илрүүлсэн Outlier - г 1 гэдэг утгаар илэрхийлж багана болгон үндсэн дата руу нэмж feature үүсгэсэн. Тухайн датаг sklearn сангийн DecisionTree алгоритмын аргаар боловсруулж машинаа сургаж ашигласан. Сургасан машин нь анх удаа харж байгаа датан дээрээ 99.99% магадлалтай outlier - ыг ангилж чадсан тул энэхүү машин сургалтыг амжилттай болсон хэмээн дүгнэж байна.