
# Delhi Weather Prediction

24.11.2019

Devans Somani (MT2019032)

Dhameliya Vatsalkumar (MT2019033)

Dhruvinkumar Radadiya (MT2019035)

## Problem Statement

We have been given a dataset which contains weather data for New Delhi, India. The dataset contains various features such as temperature, pressure, humidity, rain, precipitation,etc from 1997 to 2016. The main target is to develop a prediction model accurate enough for predicting the weather.

## Description of the dataset

We obtained our data from Kaggle. Originally this data was taken out from Wunderground with the help of their easy to use api. Wundergrounds API provides hourly weather data in JSON format. This dataset contains more than 100,000 samples, each sample having 20 features :

1. datetime_utc: Date and time of day (EST)
2. tempm: Temperature in Celcius
3. dewptm: Dewpoint in Celcius
4. hum: Humidity %
5. wspdm: Wind speed in kph
6. wgustm: Wind gust in kph
7. wdird: Wind direction in degrees
8. wdire: Wind direction description
9. vism: Visibility in Km
10. pressurem: Pressure in mBar
11. windchillm: Wind chill in Celcius
12. heatindexm: Heat index Celcius
13. precipm: Precipitation in mm
14. fog: Boolean
15. rain: Boolean
16. snow: Boolean
17. hail: Boolean
18. thunder: Boolean
19. tornado: Boolean
20. conds: Different weather conditions(Haze, Smoke, Mist etc)

## Data Visualization

- We found that the given dataset contains 38 different weather condition labels, which we will predict using the features which are given. Many weather conditions predominate other conditions which occur fewer times. The frequency of each weather condition is given in Figure 1.

```
Haze                              47602
Smoke                             20760
Mist                               9375
Clear                              3129
Widespread Dust                    2856
Fog                                2760
Scattered Clouds                   2209
Partly Cloudy                      2091
Shallow Fog                        1860
Mostly Cloudy                      1537
Light Rain                         1302
Partial Fog                        1031
Patches of Fog                      901
Thunderstorms and Rain              486
Heavy Fog                           421
Light Drizzle                       414
Rain                                394
Unknown                             383
Blowing Sand                        378
Overcast                            326
Thunderstorm                        192
Light Thunderstorms and Rain        176
Drizzle                             112
Light Fog                            64
Light Thunderstorm                   64
Heavy Rain                           28
Heavy Thunderstorms and Rain         22
Thunderstorms with Hail              11
Squalls                               6
Light Sandstorm                       6
Light Rain Showers                    5
Light Haze                            4
Volcanic Ash                          4
Funnel Cloud                          2
Rain Showers                          2
Sandstorm                             2
Light Hail Showers                    1
Light Freezing Rain                   1
Heavy Thunderstorms with Hail         1
Name: conds, dtype: int64
```

**Figure 1**

- It shows the relation between average temperature occuring in a particular month each year.

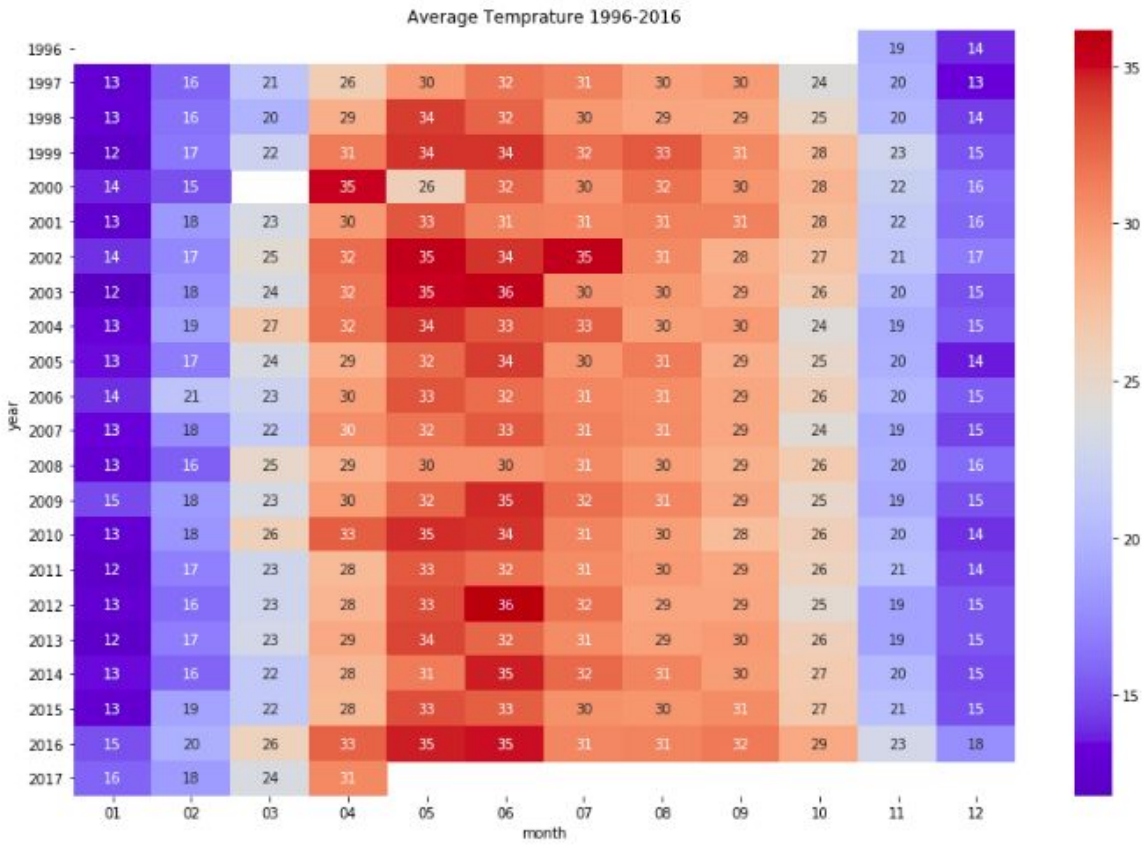


**Figure 2**

## Data Preprocessing

- Many features of the dataset contains missing values. The count of missing values for each feature is given in Figure 3.

```
datetime_utc         0
conds               72
dewptm             621
fog                  0
hail                 0
heatindexm       71835
hum                757
precipm         100990
pressurem          232
rain                 0
snow                 0
tempm              673
thunder              0
tornado              0
vism              4428
wdird            14755
wdire            14755
wgustm           99918
windchillm      100411
wspdm             2358
year                 0
month                0
dtype: int64
```

**Figure 3**

We observed that heatindexm, windchillm, wgustm and precipm features have more than 70% of there values as missing values therefore we have dropped the above stated features.

To fill the missing values of the remaining features, which have numeric data we have replaced them with their respective yearly mean/median values depending on the values which gave better accuracy, and features having string data type is replaced by yearly most frequent value.

- From the feature datetime_utc we have extracted year and month.
- We have applied scaling(StandardScaler).
- We applied PCA(Principle Component Analysis) to find out which components are more relevant.
- The feature ‘wdire’ contains non-numeric values, therefore we used One-Hot Encoding.

# Models

We used five different types of classifier-

## I. Stochastic Gradient Descent

Stochastic Gradient Descent (SGD) is a simple yet very efficient approach to discriminative learning of linear classifiers under convex loss functions such as (linear) Support Vector Machines and Logistic Regression. Stochastic gradient descent (often abbreviated SGD) is an iterative method for optimizing an objective function with suitable smoothness properties (e.g. differentiable or subdifferentiable). It can be regarded as a stochastic approximation of gradient descent optimization, since it replaces the actual gradient (calculated from the entire data set) by an estimate thereof (calculated from a randomly selected subset of the data). Especially in big data applications this reduces the computational burden, achieving faster iterations in trade for a slightly lower convergence rate.

## II. Logistic Regression

Logistic Regression is a Machine Learning classification algorithm that is used to predict the probability of a categorical dependent variable. In logistic regression, the dependent variable is a binary variable that contains data coded as 1 (yes, success, etc.) or 0 (no, failure, etc.). In other words, the logistic regression model predicts P(Y=1) as a function of X. Logistic Regression is a Machine Learning classification algorithm that is used to from sklearn.linear model import Logistic Regression.

## III. Kth Nearest Neighbour Classifier

k-nearest neighbors algorithm (k-NN) is a non-parametric method used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression. In k-NN classification, the output is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor. In k-NN regression, the output is the property value for the object. This value is the average of the values of k nearest neighbors. k-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification.

## IV. Support Vector Machine Classifier

Support Vector Machines is considered to be a classification approach, but it can be employed in both types of classification and regression problems. It can easily handle multiple continuous and categorical variables. SVM constructs a hyperplane in multidimensional space to separate different classes. SVM generates optimal hyperplane in an iterative manner, which is used to minimize an error. The core idea of SVM is to find a maximum marginal hyperplane(MMH) that best divides the dataset into classes. It is known for its kernel trick to handle nonlinear input spaces.

## V. Decision Tree

Decision tree is a type of supervised learning algorithm (having a pre-defined target variable) that is mostly used in classification problems. It works for both categorical and continuous input and output variables. In this technique, we split the population or sample into two or more homogeneous sets (or sub- populations) based on most significant splitter / differentiator in input variables. Scikit learn contain class which implement decision tree classifier. We can regulate the performance of decision tree by tuning various parameter.

| No. | Model | Accuracy |
|---|---|---|
| 1 | Stochastic Gradient Descent | 68 |
| 2 | Logistic Regression | 71 |
| 3 | k-nearest neighbors | 75 |
| 4 | Support Vector Machine Classifier | 77 |
| 5 | Decision tree | 78 |

**Table 1.**

## Conclusion

We have explored a wide spectrum of possible classifiers that might be a good fit for solving the Delhi Weather Prediction. We achieved the best accuracy using decision tree which was 78%. The reason why the accuracy is less than 80% even after trying all other models is because there are 38 different weather conditions, 26 of which doesn't even occur 1% of the time in the whole dataset, therefore any model which is trained using few samples will be biased and cannot predict the outcome properly.