Name: Xin Qin

Section: CSE 140 AC

Precipitation Prediction

1. Summary.
   a. If we have a set of previous weather data and precipitation, how could we predict the future precipitation given data;
   b. How big should the training data size be so that we could predict the data as precisely and stable as possible?
2. Motivation.

   In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under gravity, including rain, sleet, snow, graupel, and hail. The precipitation (abbreviation precip) is a very important factor for agricultural economics, local transportation, and people traveling. If we do not have a good precipitation prediction, then not only people's daily life, but also local economics would be negatively affected. If there exists a relatively accurate way to predict precipitation, then more conveniences will be provided. For example, people could look up the precipitation for some day to decide if they would travel, or even if they should bring umbrella; farmer would adjust the amount of water for their crop if they could know the future precipitation; local department of transportation could give out warnings on some roads or close dangerous roads according to the prediction of precipitation. Thus, the prediction for precipitation is very important.

   In the process of prediction, it would be great if the prediction result were accurate. However, it is not possible because it is prediction. There must be some errors. Letting people directly see the difference between prediction data and the actual value is very important to express the performance of the prediction. Besides, we also need to know what number of training data is large enough so that we could relatively rely on it. All in all, we need to find a way to compute the possible value of precipitation with previous weather data and visualize the errors of the prediction.
3. Dataset

   For dataset, I am using "Weather Underground Weather Data". After opening the website, we could see a time bar at the top of the form. You can manually choose the time period for the weather data that you prefer to use. My time period of data is from January 1st 2002 to March 1st 2013. The thing that should be cared about is that the website could only show the data for one year once, so if we need the data of more than one year, we have to download several times. After choosing the time period, we can scroll down the webpage and then we can download these data in csv form by simply clicking on "Comma Delimited File" and saving as "webpage, HTML only".

4. Algorithm
    a. For the first problem, we assume that we have a large, well-form set of previous weather data, including single day's max temperature, mean temperature, min temperature, max dew point, mean dew point, min dew point, max humidity, mean humidity, min humidity, max sea level pressure, mean sea level pressure, min sea level pressure, max visibility miles, mean visibility miles, min visibility miles, max wind speed, mean wind speed, max gust speed, cloud cover, and precipitation. Specifically, I chose Seattle as my location. In this data set, precipitation is a float number so it is not a good strategy to directly treat the value of precipitation as labels to train and predict. So I split all the possible value of precipitation into five classes: 0.0 (class 1), 0.0 ~ 0.2 (class 2), 0.2 ~ 0.5 (class 3), 0.5 ~ 1.0 (class 4), > 1.0 (class 5). Other elements except the precipitation will be gathered as a feature vector v_i (the dimension is 19). The first step is to compute the mean vector and standard deviation vector for $v_1, v_2, \ldots\ldots, v_n$, call them as m and s, respectively. Then compute $v_i = \frac{v_i - m}{2 * s}$ to process all the feature vectors. Finally, I will feed all the features vectors and their corresponding label into supported vector machine (SVM) and get a linear model from it. Every time when I get a feature vector that needs to be predicted the label, I will use the linear model that I already get from the SVM to predict the result of the input feature vector.
    b. In order to let people directly see the performance of the prediction, I plot histogram for each prediction case, containing two bars: one is the actual value; the other one is the predicted value. To see how the accuracy is affected by the size of training data, I plot two images. One is a linear graph presenting the relationship between accuracy and data size. The other one is also a linear graph that presents the relationship between the standard deviation of accuracy and data size.
5. Result:
   With a large set of data, I get a linear model about the relationship between the precipitation and the other weather conditions. With the linear model I can predict the future precipitation and due to the known data of precipitation and the predicted precipitation, I can calculate the accuracy. Eventually, I found out that there is not an exact relationship between the accuracy and training data size. I think it is because every time when I selected a set of data to train, I selected them randomly and the performance is highly depend on those selected data rather than how much data I used to train. So I calculate the stable of predicting with specific amount of training data represented by the standard deviation. I plot the standard deviation and see that it is more stable with larger training data set.
6. Reproducing the result:
   In the files I submitted, there are four files that support the libSVM (supported vector machine): libsvm.dll, libsvm.so.2, svm.py, svmutil.py. These four files are downloaded from the libSVM website. I simply changed the loading paths in these files and then I

import then in my python files. There are three files: predict-data.txt, predict-small.txt, train-data.txt. The first one is for testing the performance of the program, the second one is mainly for plotting the prediction result (you could also plot the prediction result from predict-data.txt, but it would be very messy plot), the third one is the training data, it includes ten-year weather data that I download online. There are two more files: precipitation-prediction.py and test.py. The precipitation-prediction.py accept three arguments: size of training data (from 1 to 4000, 0 is for using full training data set), the prediction file name containing data to be predicted, and an option for drawing plot (if you want it to produce a plot, simply invoke 'y', invoke anything else otherwise). For example: python precipitation-prediction.py 3000 predict-small.txt y. This means using 3000 data to train, predict the data stored in predict-small.txt, and draw the plot. It will also print out the accuracy of prediction for the current training and predicting. The test.py is a python script to test the performance of precipitation-prediction.py. You could just type in the command line: python test.py. It will produce two plots: performance.jpg and standard-deviation.jpg. The first image is the mean accuracy of prediction with different data size whereas the second image is the standard deviation of accuracy of prediction with different data size. The execution of test.py may take long time (several minutes) because the training of svm costs ten to twenty seconds once and I run the precipitation program with 12 different data size, 20 times for each. If you want to change the times of testing, you could simply open test.py, change the value of sizes and DEFAULT_TIME. I tested the entire program on my Ubuntu Linux machine, and it works fine. It should also work on windows machine and Mac but I am not sure. So I suggest it is best to be tested on Ubuntu Linux machine.

By the way, the reason I did not directly test the performance of svm in precipitation-prediction.py is that if I run the svm_train multiple times at the same time, the result of prediction will be the same not matter what data it predicts. So I separate the mainline of precipitation prediction and test into to python files.

7. Collaboration:

   None

8. Reflection:

   When I was finishing this assignment, I used the knowledge I learned from previous homework and my knowledge from Statistics to predict and interpreted the change of weather. From this assignment, I realized that Statistics can combine with data computing and with that we can predict the future weather and even more. I wish I could remember more from statistics class so that I do not have to spend that much time doing formula search.