ML Project

Determining whether weather is good for picnic

Author: Marián Kravec

Data

a) Source and content

Data used for classification and prediction are from Kaggle (`https://www.kaggle.com/datasets/sujaykapadnis/whether-prediction-dataset?select=weather_prediction_dataset.csv`).

Dataset contains weather information from 18 different European cities (or places) measured between years 2000 and 2010 (3654 rows).

We will use data for 16 of these cities. We will not use data for Roma because it's missing picnic weather label and data for Malmo because it contains only 6 out of 12 columns.

We have these columns for specific cities:

	cloud cover	wind speed	wind gust	humidity	pressure	global radiation	precipitation	sunshine	temp mean	temp min	temp max	picnic weather
De Bilt	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Tour		✓		✓	✓	✓	✓		✓	✓	✓	✓
Ljubljana	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓
Maastricht	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Muenchen	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Perpignan		✓		✓	✓	✓	✓		✓	✓	✓	✓
Heathrow	✓			✓		✓	✓	✓		✓	✓	✓
Budapest	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓
Montelimar		✓		✓	✓	✓	✓		✓	✓	✓	✓
Dusseldorf	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Dresden	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓
Stockholm	✓				✓		✓	✓	✓	✓	✓	✓
Sonnblick	✓			✓		✓	✓	✓	✓	✓	✓	✓
Basel	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓
Kassel		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Oslo	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

b) Data preparation

Firstly we need to prepare data into suitable form for models. We split data into city specific dataset.

Then when we are preparing data for model we load data for required city (or multiple cities) and shuffle rows. Then we split data firstly into variables and labels (meaning column-wise) and secondly into train, test and validation parts (meaning row-wise). Finally for train data we compute average and variance of each column to normalize all three datasets using these values.

SVC comparison

We will try to classify whether weather is suitable for picnic in Basel. We will train three Support vector classifiers to compare their results.

We will train these three models:

- C-Support Vector Classifier (sklearn.svm.SVC) with radial basis function kernel (we will optimize regularization parameter C)
- Linear Support Vector Classifier (sklearn.svm.LinearSVC) (we will optimize regularization parameter C)
- Linear SVC with SGD (stochastic gradient descend) training (sklearn.linear_model.SGDClassifier with loss parameter "hinge") (we will optimize constant that multiplies the regularization term α)

As a score for each model we will take percentage of correctly assigned labels. To make this value more informative we will use cross validation and final score would be average score of 5-fold cross validation.
After training (and optimizing chosen parameters) we get these test scores:

model	optimized parameter	parameter value	test score
RBF kernel SVC	C	23	0.89315
Linear SVC	C	23	0.94247
Linear SVC with SGD	α	10^{-8}	0.89315

Based on this information it seems like Linear SVC without SGD is better then other two models which seems to equally good. However as Goodhart's law says "When a measure becomes a target, it ceases to be a good measure"which means that test score is not good measure for our models (test dataset was used to choose best parameter value).

If we compute validation score for each of these models:

model	validation score
RBF kernel SVC	0.92632
Linear SVC	0.95361
Linear SVC with SGD	0.88804

We can see that Linear SVC still has best score, so we can consider it to be best model out of these three. We can also see that based on validation score it seems like SVC with RBF kernel is better model than Linear SVC with SGD.

Basic neural network versus transfer learning

We have 5 cities for which we have all 12 columns, those cities are: De Bilt, Maastrich, Muenchen, Dusseldorf, Oslo. What we will try to do is train standard sequential neural network to classify whether weather in Maastrich is good for picnic.

We will compare two neural network. First NN will be trained solely using data from Masstrich. Second will be firstly trained on data from four other cities then all layers except last one will be frozen and last layer would be retrained using Masstrich data. We will then compare those two networks.

We tried multiple neural networks with different number of layers and different number of neurons in each layer and finally we ended sequential neural network with four dense layers. First three of these use hyperbolic tangent as activation function and have 24, 8 and 12 neurons respectively. Last layer use softmax as activation function and contain 2 neurons representing true and false values of our labels.