Predicting Peak Bloom Date of Cherry Trees

Supervised Multi-class Classification Models: SVM, RNN (LSTM)

Yanyu Chen, Zihan Zhang, Hongyi Zheng

May 11th, 2021

CSCI-SHU 360 Machine Learning

Outline

1. Introduction

- Cherry Blossom
  - Widely distributed in the Northern hemisphere. Japan, China, Russia, United States, etc.
  - Ornamental purpose: “In 2018, an estimated **63 million** people travel to and within Japan… Spending around **$2.7 billion**… More than **40%** of foreign visitors” (The National News)
- Stimulus & Aspiration
  - Novelty of idea.
  - Provide tourism guidance (manage schedule).
  - Pollen season alerts.
  - Possibly inspire agricultural planting and induce financial benefits.
- Objective & Contribution
  - Given past sequential daily average temperatures records, implement ML techniques to predict the future exact peak blossom date of Cherry trees.
  - **Multi-class classification problem**
    - SVM: Non-sequential time interval
    - LSTM RNN: Interaction between different specific timestamp in sequential series

2. Related Works

- Aiying Zhang, Huanjiong Wang
  - Among all meteorological features, daily average temperature correlates to the first flowering date and full flowering date of ornamental plants (Magnolia, Subhirtella) in the Beijing area most strongly through statistical analysis.
  - **Accumulated temperature** (> certain threshold): describe the growing process of plants.
  - Adding other factors, like relative humidity, solar radiation, wind speed measurement, might improve the prediction accuracy.
- Jenny Cifuentes, Geovanny Marulanda
  - Review the different machine learning strategies for (hourly) temperature forecasting.
  - "Support Vector Machines are preferred based on their good compromise between simplicity and accuracy."
- Chunqiao Mi, Jianyu Yang
  - Prediction of accumulated temperature in vegetation period in Northeast China.
  - "Artificial neural network (back-propagation algorithm) are more applicable than regression models when predicting accumulated temperature".

3. Dataset & Preprocessing

Select date intersection	Full-Flowering (>70%) Date	Historical Series of Phenological data
Washington, D.C., United States	United States Environmental Protection Agency: 1921-2016 around Tidal Basin	U.S. National Oceanic and Atmospheric Administration
Kyoto, Japan	Osaka Prefecture University: since 810 AC (Only select 1881 - now)	Japan Meteorological Agency

- ❖ Famous tourism cities with comparable geographical features (similar latitude, coastal).
- ❖ Date format: M-D-Y → Date of Year = DOYS (int type).
- ❖ Preliminary cleaning: fill missing data (e.g. average tmp = (low + high) / 2) and standardize the dimension.
- ❖ SVM's Training Data (unbalanced):

$$\text{Train data} = (X_{vec}, y_{label}) = (X_i, y_i)$$

$$X_{vec} = X_i = \text{Array of 10 days (TOBS) before the interested date.}$$

$$y_{label} = y_i \in \{0, 1, 2, ..., 9, 10\}$$

$$y_i = 0 \rightarrow \text{Cherry tree will not blossom in future 10 days.}$$

$$y_i = k \rightarrow \text{Cherry tree blossoms in } k \text{ day(s) later.}$$

4. Model 1: Multi-class SVM

- **4.1 Methodology**
    - Multi-class approach: OVO / OVR ? → Our choice & reasoning
    - Primal Problem & Kernel Trick → Our reasoning
    - Imbalance data → Our solutions
- **4.2 Evaluation & Discussion**
    - Ordinary SVC | Weighted SVC | Oversampled SVC
    - Confusion Matrix & PR-curves

4.1.1 Why OVO?

- Our Choice: **One-Vs-One strategy** (OVO)
- Reasoning:
  - No ambiguous region, more generalized classifier
  - Less ill-conditioned / More independent than **One-Vs-Rest (OVR)** *(Pornntiwa Pawara, “One-vs-One classification for deep neural networks, 2020”)*

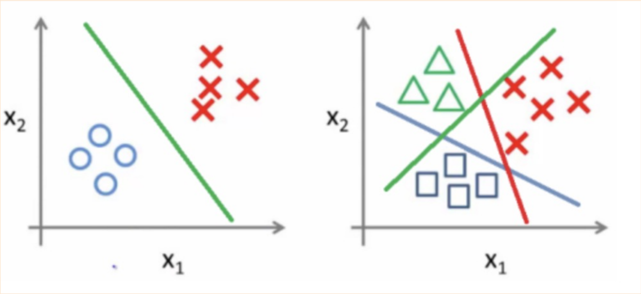

Fig 1: Binary Classification vs. Multi-class Classification

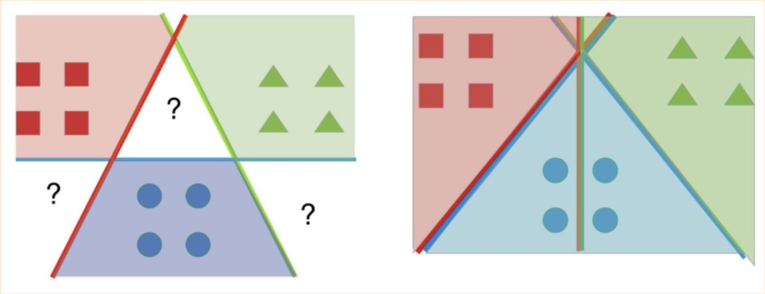

Fig 2: Seperation with OvR

Seperation with OvO

4.1.2 Objective

- **Primal Problem Equation**

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \quad \underbrace{\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}} + \underbrace{C\sum_{i=1}^{l}\xi_i}$$

$$\text{subject to} \quad y_i(\boldsymbol{w}^T\phi(\boldsymbol{x}_i)+b) \geq 1-\xi_i,$$
$$\xi_i \geq 0, i = 1,\ldots,l,$$

- **RBF Kernel trick**
  - More flexible decision boundary
  - good for non-linear data

$$k\left(\mathbf{x},\mathbf{x}'\right) = \exp\left(-\gamma\|\mathbf{x}-\mathbf{x}'\|^2\right)$$

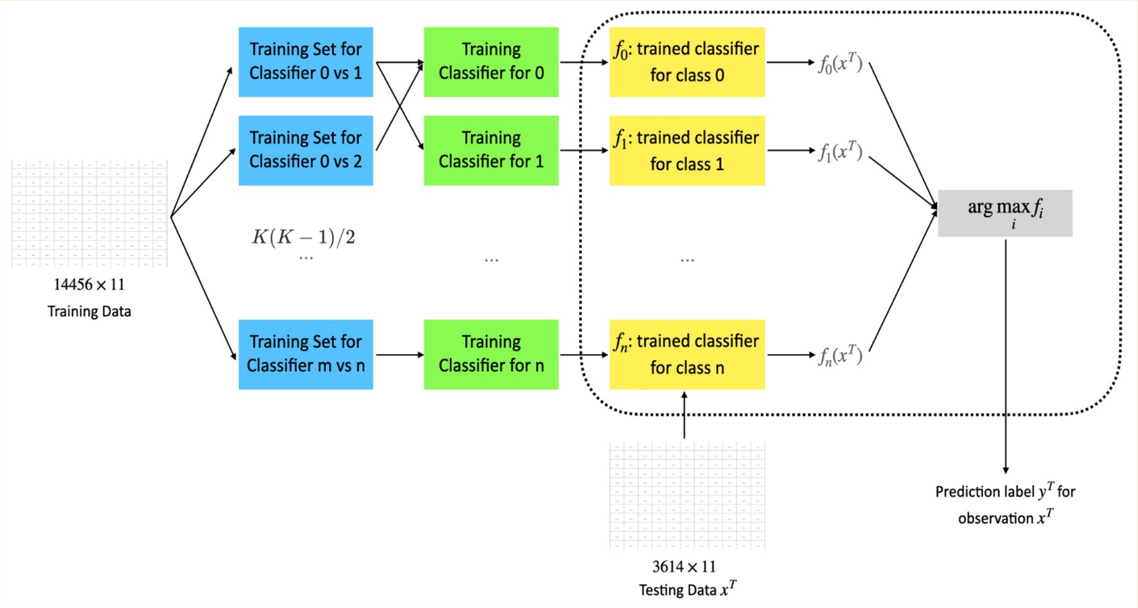

Fig 3: OvO Multi-class SVC Flow Diagram

4.1.3 Imbalance Data & Solutions

- **Imbalance problem**: (common)
  - Favor majority class, useless classifier
- **Solution 1**: Alternating penalization weights of different classes <u>proportionally</u> (sklearn-SVC: “balanced”)

$$\begin{aligned} \min_{\boldsymbol{w},b,\boldsymbol{\xi}} \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C^{+}\sum_{y_i=1}\xi_i + C^{-}\sum_{y_i=-1}\xi_i \\ \text{subject to} \quad & y_i(\boldsymbol{w}^T\phi(\boldsymbol{x}_i)+b) \geq 1-\xi_i, \\ & \xi_i \geq 0, i=1,\ldots,l, \end{aligned}$$

$$w_j = \frac{n}{kn_j}$$

$$C_j = C * w_j$$

- **Solution 2**: <u>Oversample</u> the training data proportionally (Imblearn-SMOTE)

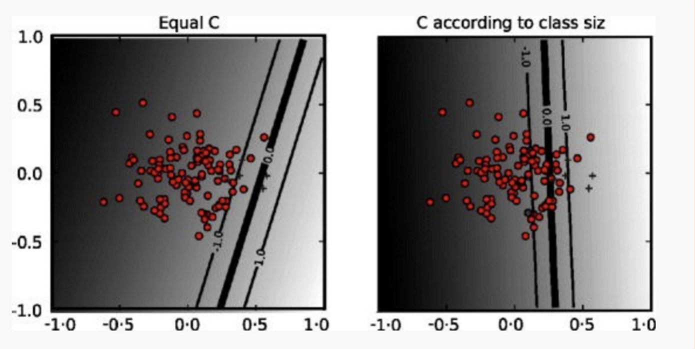

Fig 4: Different Penalization Effects

4.2.1 Evaluation

Evalutation Mertic	Value
SVM Accuracy Score	87.1562932226833
SVM F1 Score	56.8725470570625%
SVM Precision Score	43.5066551861651%
SVM Recall Score	93.4460037087899%

Table 1: Ordinary SVC (clf) Result

Evalutation Mertic	Value
SVM Accuracy Score	88.8603042876902
SVM F1 Score	74.4670663962163%
SVM Precision Score	78.2935254240545%
SVM Recall Score	71.2101219882177%

Table 2: Weighted SVC (wclf) Result

Evalutation Mertic	Value
SVM Accuracy Score	98.8726371718234
SVM F1 Score	83.2817162723723%
SVM Precision Score	86.1213123738272%
SVM Recall Score	79.8827261612734%

Table 3: Oversampled SVC (oclf) Result

- Accuracy **X** → Precision & Recall YES!
- Different distribution of weight improves Classifier -- Expectation matched
- Multi-SVC less prone to majority class

4.2.1 Evaluation

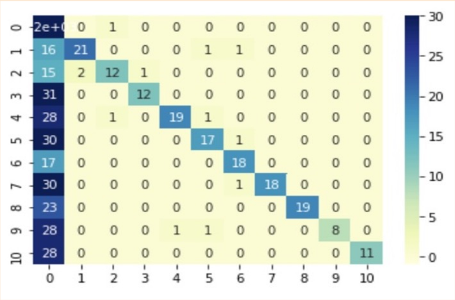

Fig 5: **Ordinary** SVC (clf) Confusion Matrix

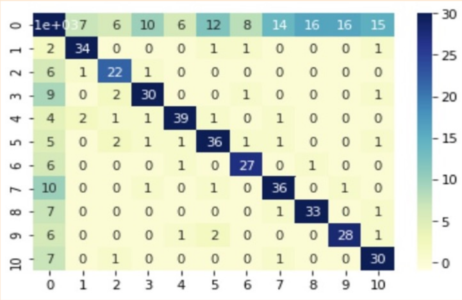

Fig 6: **Weighted** SVC (wclf) Confusion Matrix

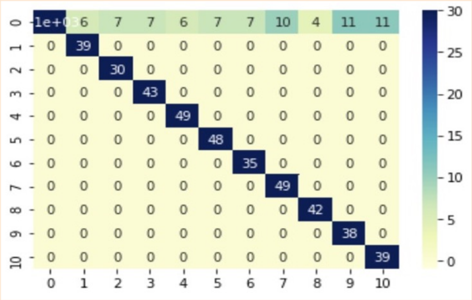

Fig 7: **Oversampled** SVC (oclf) Confusion Matrix

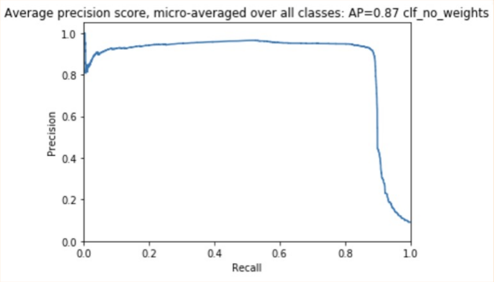

Fig 8: Ordinary SVC (clf) PR Curve

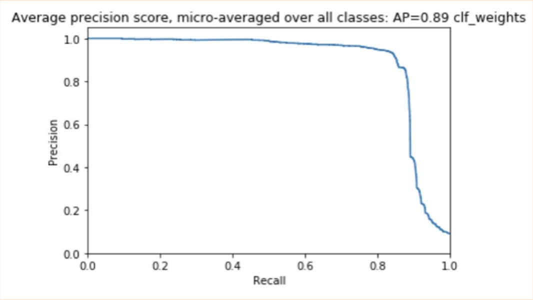

Fig 9: Weighted SVC (wclf) PR Curve

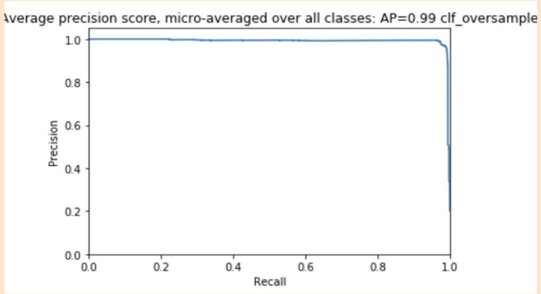

Fig 10: Oversampled SVC (oclf) PR Curve

4.2.2 Discussion

- No K-fold cross validation. (8-2-Division: Andrew. Ng)
- Temperature is not the only features: Humidity, wind etc.
- Hard to generalize/apply in reality
- Let’s explore Neural Network!

5. Model 2: LSTM for Multi-classification

- **5.1 Methodology**
    - Sliding-window approach → Our choice & reasoning
    - Comparison Between Classification and Regression
    - Model Structure
- **5.2 Evaluation & Discussion**
    - Classificataion Approach

5.1.1 Sliding Window Approach

- Several ways to train our LSTM model:
  - **Many to Many**: Use all temperature data within a year as the input sequence and get predictions for each date at once
  - **Many to One**: Use the sliding window approach to get temperature data within a fixed number of days before each certain date as the input and get predictions for each date separately ✅
- Advantages:
  - relatively short RNN chain, allow faster training
  - Fixed sequence length, much easier to implement

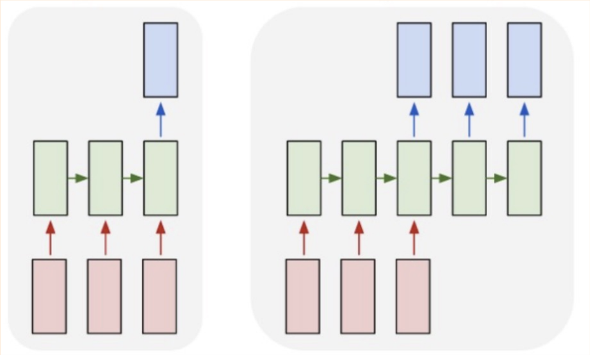

Fig 9: Many-to-one vs. Many-to-

5.1.2 Classification v.s. Regression

Output

- Classification problem
  - The output would be k + 1 classes (0, 1, 2, …, k).
  - y = 0 representing the peak blossom date is more than k days away from the current date,
  - y = i (i ≠ 0) representing the peak blossom date is “i” days away.
- Regression problem
  - The model will directly output the number of days between now and the estimated peak blossom date.

Model Sturcture

Our LSTM module is a double-layer model with input sequence length equals to the number of dates we consider temperature from (e.g. 10, 20 or 30), the output is then processed with a softmax layer to output a vector of probability of each classes.

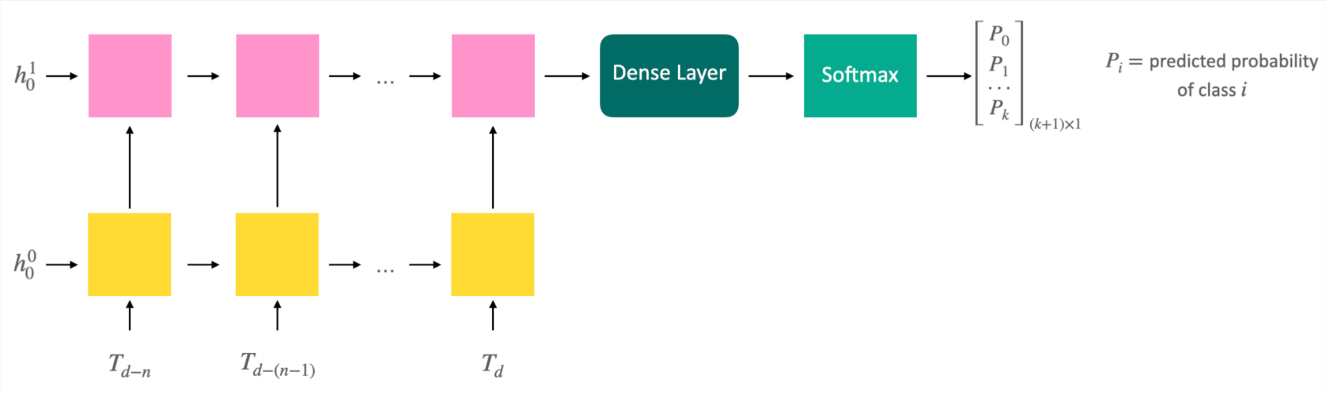

Fig 10: LSTM Pipeline

5.2.1 Evaluation

Due to the extremely long training time of the LSTM model, it is not possible to generate a PR curve, so we will directly report the result of a single model.

Evalutation Mertic	Value
Loss	2.3103382587
Precision	19.9258928001%
Recall	13.9103545761%
F1 Score	14.3619214968%

Table 4: LSTM Result

- The result is based on the data after oversampling
- num_layer = 2, input_size = 20, hidden_size = 30, dropout = 0.5

6. Conclusion

- **Limitation**
  - Imbalanced data (in label 0).
  - Not considering other meteorological features that may provide a more comprehensive description of regional climate.
  - Multivariable/Longer sequential data is time-consuming to be trained in the LSTM model due to CPU/GPU limitation.
- In our case, **SVM** classification performs **better** than LSTM RNN.
- In the case of the **LSTM** model, the **regression** approach might outperform the **classification** approach.


Thank you for your listening~~