CS 4780/5780: Machine Learning for Intelligent Systems **(Due: 5/13/20)**

Final Project: COVID-19 Hospitalizations Prediction for EU Countries

Instructor: Thorsten Joachims

Course Policy: Read all the instructions below carefully before you start working on the final project, and before you make a submission.

1 Introduction

The final project is about conducting a real-world machine learning project on your own, with everything that is involved. Unlike in the programming projects 1-5, where we gave you all the scaffolding and you just filled in the blanks, you now start from scratch. The past programming projects provide templates for how to do this, and the most recent video lectures summarize some of the tricks you will need (e.g. feature normalization, feature construction). So, this final project brings realism to how you will use machine learning in the real world.

The task you will work on is predicting hospitalizations due to COVID-19. Having good predictions is crucial for being prepared (e.g. getting supplies, planning staffing), and it is not be hard to see that your machine learning expertise can have crucial impact in the world. Although hospitalizations are directly related to COVID-19 cases, the different populations, timelines and policy interventions of different EU countries result in different trends in hospitalization numbers. In this project you will be making predictions for the country-level hospitalizations using COVID-19 age group case data and also previous hospitalization data. There will be two tasks, one will be a basic problem that will require you to use methods learned in class. The second task will be more difficult and will require some additional intuition and insight.

2 Dataset

The dataset you will be working with in this project is the EU country-level COVID-19 cases by age group and hospitalizations. We have split the data into a training and testing set in a specific way and we will give them to you as separate files for each task. You will be provided with the following files:

- **train_baseline.csv**: This is the dataset containing information about the country-level COVID-19 cases and hospitalizations. The columns are country, date, year & week, daily hospitalizations, weekly new cases for ages under 15, ages 15-24, ages 25-49, ages 50-64, ages 65-70, and ages 80 and over. Finally, the last columns are binary labels that are 1 if the next week hospitalizations increase and 0 if they decrease. This is what you will need to predict for the test points.
- **train_creative.csv**: This is the same data as **train_baseline.csv**, except the last column has the number of hospitalizations one week out instead of the binary value. This is what you want to predict in the creative task regression instead of the binary labels.
- **train_creative_t_values.csv**: This dataset contains extra information to be used (if needed) in the creative problem. The first three columns are the same as **train_creative.csv**, but the next 50 columns are the t-minus values that are the hospitalizations t days before the current date. (ex. column t-10 is the number of hospitalizations 10 days ago). This is a lot of data so feel to use any number of these columns to improve your prediction.
- **test_baseline_no_label.csv**: This is the dataset with the same columns as **train_baseline.csv** for the test months, except that the next week hospitalizations increase/decrease column which imply the target true labels for the task have been omitted. Note that you are not predicting the hospitalization counts for the baseline, but the labels indicating an increase or decrease. Your prediction to be submitted to Kaggle will be the binary prediction for the dates in this file.

- **test_creative_no_label.csv**: This is the dataset with same columns as **train_creative.csv** for the test dates, except that the last column which has the hospitalization values 1 week out have been omitted. Here, you **are** predicting the actual count of hospitalizations. (This is essentially the same file as **test_baseline_no_label.csv**)
- **test_creative_t_values_no_label.csv**: This dataset contains the t-values for the points in the test set. It is the same as **train_creative_t_values.csv** except for the test dates. Again, you can combine this with **test_creative_no_label.csv** to try and improve your prediction.

3 Your Task

You will be provided with a template Jupyter Notebook and you are required to do the necessary data preprocessing (e.g. normalizing the features), model construction and selection with validation, hyperparameter tuning, and generating test-sample predictions with this Jupyter Notebook. You are encouraged, although not required, to use existing machine learning packages and frameworks for your modeling process. Examples of such packages are Scikit-learn, PyTorch, Pandas. In a fast moving field like machine learning, these packages change all the time. So, part of the realism of the final project is that you can make decisions on which packages to use, and read up on these packages, on your own.

There are two tasks, a basic and a creative. For the basic task, you are asked to give a binary prediction as to whether the number of hospitalizations goes up or down in the next week for the points in the test months in **test_baseline_no_label.csv**, which includes the features of the test dates. For the creative task, you are asked to give a prediction for the **number** of hospitalizations one week out in **test_creative_no_label.csv**, which again includes the features of the test dates. You will join both Kaggle competitions to submit CSV files with your prediction on the test dates and we will test your submission in terms of **accuracy** and **mean squared error** for the baseline and the creative respectively. (This will be explained further in the Jupyter Notebook).

4 Collaboration

You can work on the final project on your own or form a group of 2 or 3 students. You cannot discuss ideas or share code with other groups. If you are stuck in any part for a long time, please feel free to discuss your issue with the course staff during office hours. In particular, each project team will have a particular TA assigned as a mentor. This mentor is a good person to reach out to first with any questions.

5 Important Rules and Academic Integrity

The following are the important rules concerning academic integrity for this project. We will run your code to test that your submitted prediction accuracy is legitimate.

- The prediction task is for the EU COVID-19 hospitalizations, which is public information. So, you are not allowed to use outside sources that reveal any information about these statistics. So, using such external information to boost your prediction accuracy is definitely a violation of academic integrity, and your submission file to Kaggle should be generated from your notebook and not be further modified in any way.
- The only input that your training algorithm can use is the files that we provide for you, **but you CANNOT use any extra dataset resources beyond what is provided** to train your algorithm and boost your accuracy.
- Collaboration between groups is also a violation of academic integrity, and a key aspect of the project is to make your own choices for feature engineering, choose between training algorithms and tuning your models.
- In the dataset, we give you the hospitalization numbers of past dates as extra features and they are also included in the test points. However, you are not allowed to **directly** use that data to predict values for other dates. (i.e. you are not allowed to use the past numbers from tomorrow's data point to get hospitalizations today). More generally, you are not allowed to use any information from future time points as features, since in a real application this information would not be known.

It is fine to use resources like software packages and read published papers. But make sure to reference and cite any resource you used in your notebook.

6 Grading

Roughly 75% of your grade will come from how well you craft a basic solution to the learning problem. You are required to do some kind of train and validation split for model selection and provide test-sample predictions for at least two machine learning algorithms from class. In more detail, this includes:

- Load and preprocess (e.g. feature transformations, feature selection) the dataset, and design a machine learning approach to this problem (e.g. turn into multiple classification problems). Explain in words what you did and why you made these choices.

- Perform model selection via some form of train and validation split. Explain in words what you did and why you chose the methods.

- Use at least two different training algorithms from class. Explain why you made these choices.

- Reaching the baseline accuracy of 70% in Kaggle.

As you can see, writing and justifying your choices is just as important as writing the code and getting good performance on the test set.

The remaining 25% will be given for creative ideas that go beyond the basics. Again, this gives some realism to the final project, since no machine learning project is like the other, and they all require some creativity. Here are some ideas of what you could try, but there is really no limit to your creativity in improving on the basic solution.

- Create new features from the features you already have, or apply other learning algorithms that are better suited for this task, or modify some of the learning algorithms to better model this particular problem (e.g. choice of loss function to train with).

- In addition to running the experiments, clearly write up your reasoning and how each thing you tried improved - or did not improve - the results.

- Some of the ideas we tried were able to bettern than a mean squared error of 150k in Kaggle, and we will reward solutions that achieve this as well.

Clearly, you cannot do all of those, and this is not an exaustive list of things you could try. Be creative, clearly describe what you tried, and report your results in a convincing way. If you try creative and well-argued ideas that do not pan out in better prediction performance, this is a perfectly fine outcome and can be an excellent project.

We are also planning to give some extra credit for particularly accurate solutions, in particular, those that get <100k mean squared error on the private Kaggle test set or that are the among top 10 in Kaggle on the private part of the test set. Note that the test set is split to two parts in Kaggle, the score (accuracy/mean squared error) you see in Kaggle is the public part while the score on the private part of your submission is not visible until the final project is over. We will use the public accuracy for the 70% baseline and the 150k mean squared error creative solution, but for the extra credit you have to do well on the private accuracy. Hence getting into the top 10 in the public leader board does not guarantee the extra credit, which is again part of the realism of working in machine learning. The proof is in how well your method does in the real world – and the more robust your model selection, the more likely your learned rule will do well.

7 Due Date

The final project will be due on Thursday, May 13th. You need to submit your Jupyter Notebook **and** a PDF version of your notebook with answers to all questions on Canvas. **Note that May 13th is a hard deadline for this project, no late submissions will be accepted and you cannot use any of your unused slip days**.

8 Useful Resources

On Canvas, you can find a notebook as a primer on how to install your own Python environment and many useful ML packages. Furthermore, here are some official tutorials for some packages that you may find helpful for the project. Note that these packages are recommended but not required for this project.

- Pandas: https://pandas.pydata.org/pandas-docs/stable/user_guide/10min.html
- Scikit-learn: https://scikit-learn.org/stable/tutorial/basic/tutorial.html
- PyTorch: https://pytorch.org/tutorials/

9 Kaggle Competition

- Basic Solution: https://www.kaggle.com/t/1d9aa4872e5a4f7aa2d1ed2630511a44
- Creative Solution: https://www.kaggle.com/t/0541e56a59fb467b8dcf157114bcdd74