IS4242: Group Project

Oct 6 - Nov 10

Aims

- Synthesize what you have learnt so far
  - Descriptive & Predictive Analytics
- Participate in a global data science challenge
- Open ended (with some rules)

Challenge

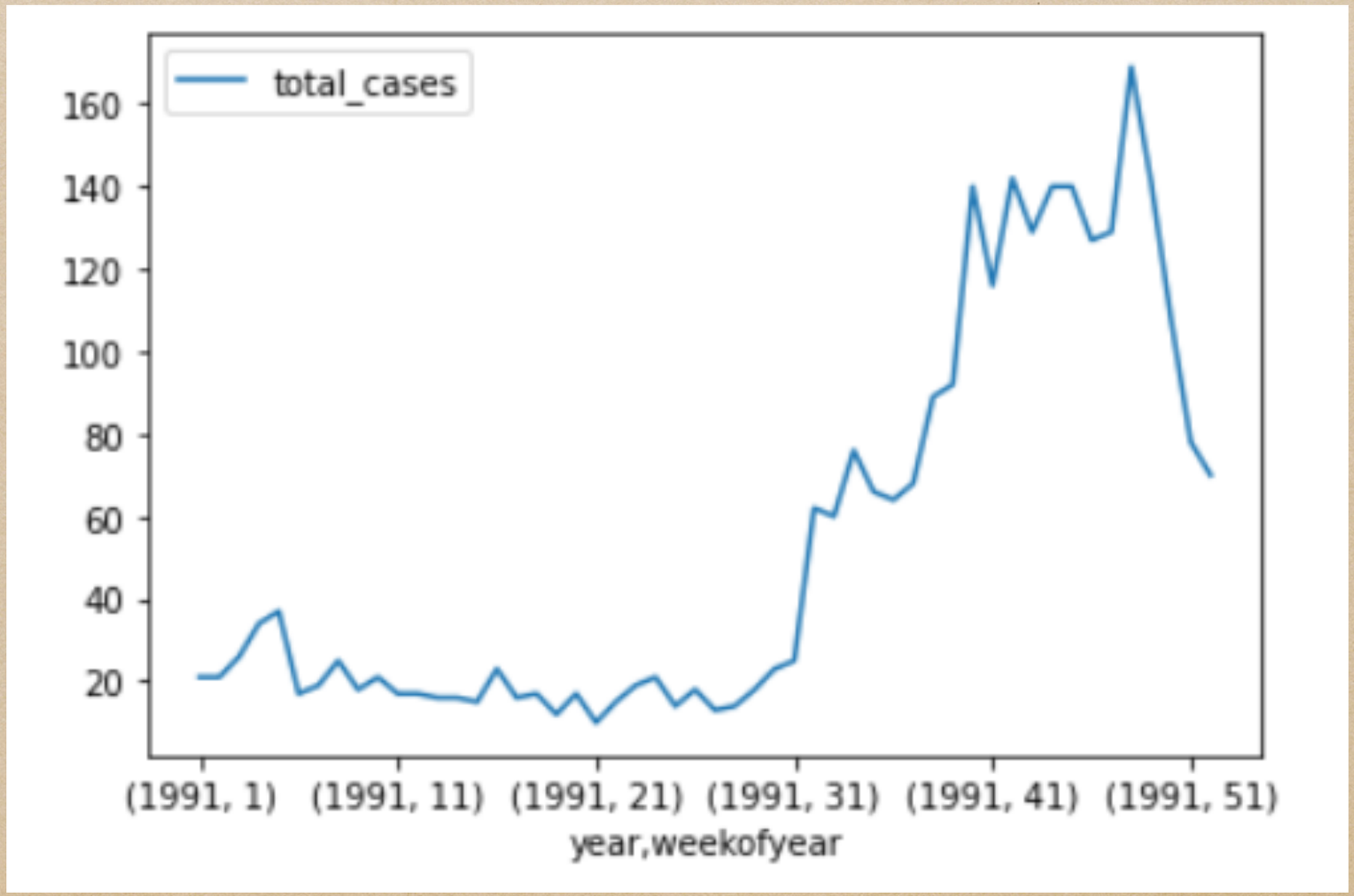
- Participate in groups of 2 (or 3) in the challenge:
  - DengAI: Predicting Disease Spread
    - Hosted by DrivenData
  - See Problem Description

Panel Data

- Observations measured repeatedly over time
- Unit of time can vary across applications (weeks in this competition)
- At each time unit, values of other features also given

Time unit	Temperature	Precipitation	#cases
15	23.5	68	5
16	24	56	5
17	27	30	3
...	...	...	...

Target Variable



Target Variable

- Number of cases
- Counts
  - Numerical but not real
  - Ordinal variable
- Ordinal Regression Problem

Schematic of Data

- Column: feature (color-coded)
- Row: time point
- Each circle: measurement (value not shown)
- There may be missing values

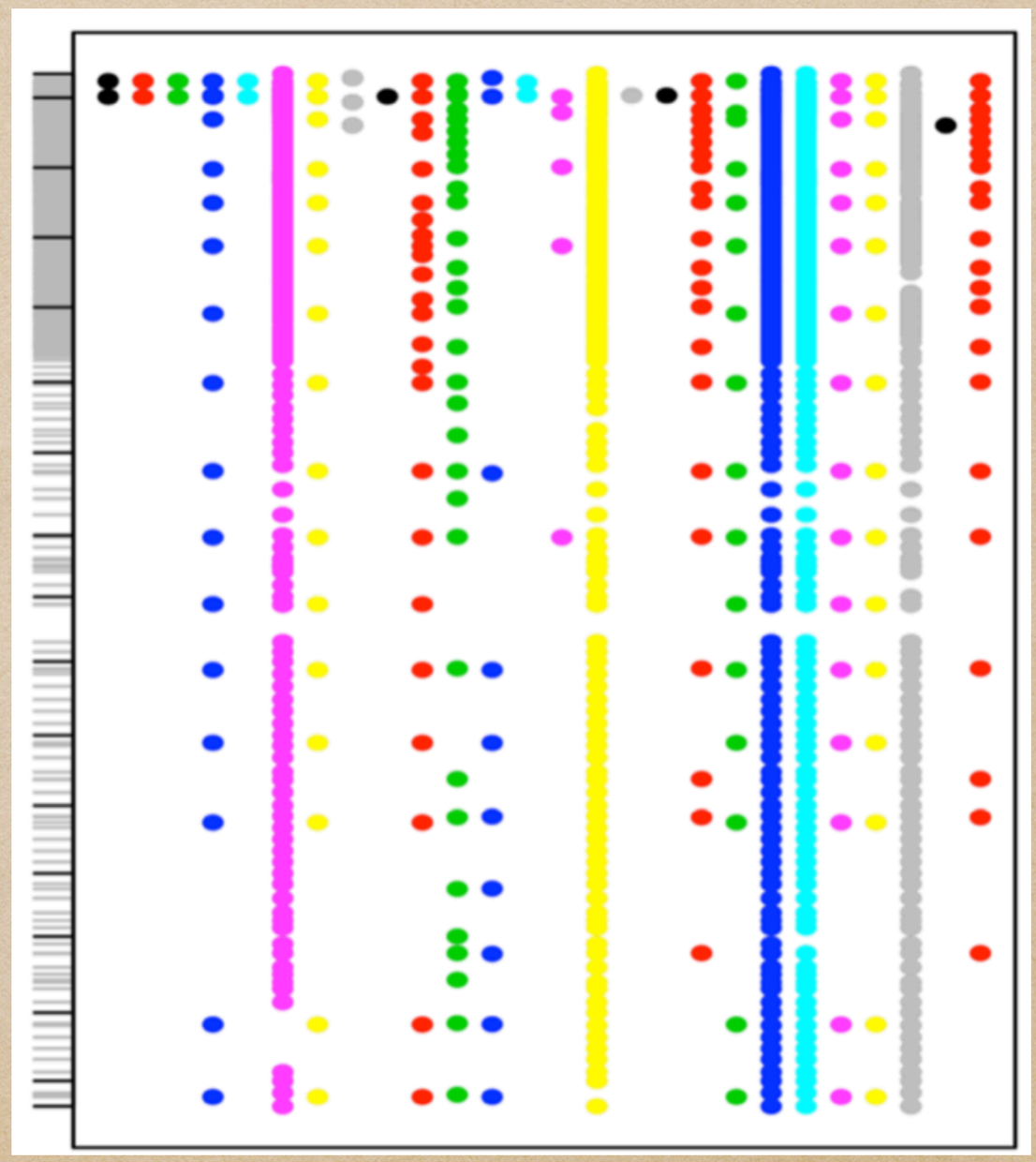
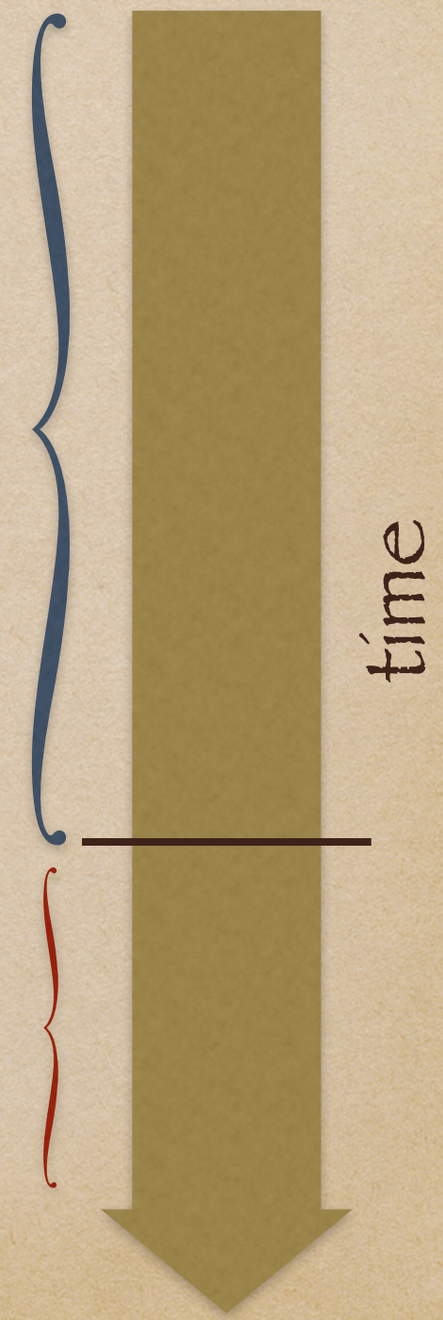


Figure: Futoma J, Hariharan S, Heller K. Learning to detect sepsis with a multitask Gaussian process RNN classifier. ICML 2017.

Train-Validation Splits

- Hold-out Validation/ Test splits have data from the future (i.e., after train data)

- Similar to real life forecasting scenario



Approaches

- 5 approaches outlined (in following slides)
- In all 5 cases you have to do data exploration - preprocessing - feature engineering - model building and evaluation steps

Approach 1

- Ignore temporal aspect
- Treat each row as independent observation - consider entire data as design matrix
- Build regression model

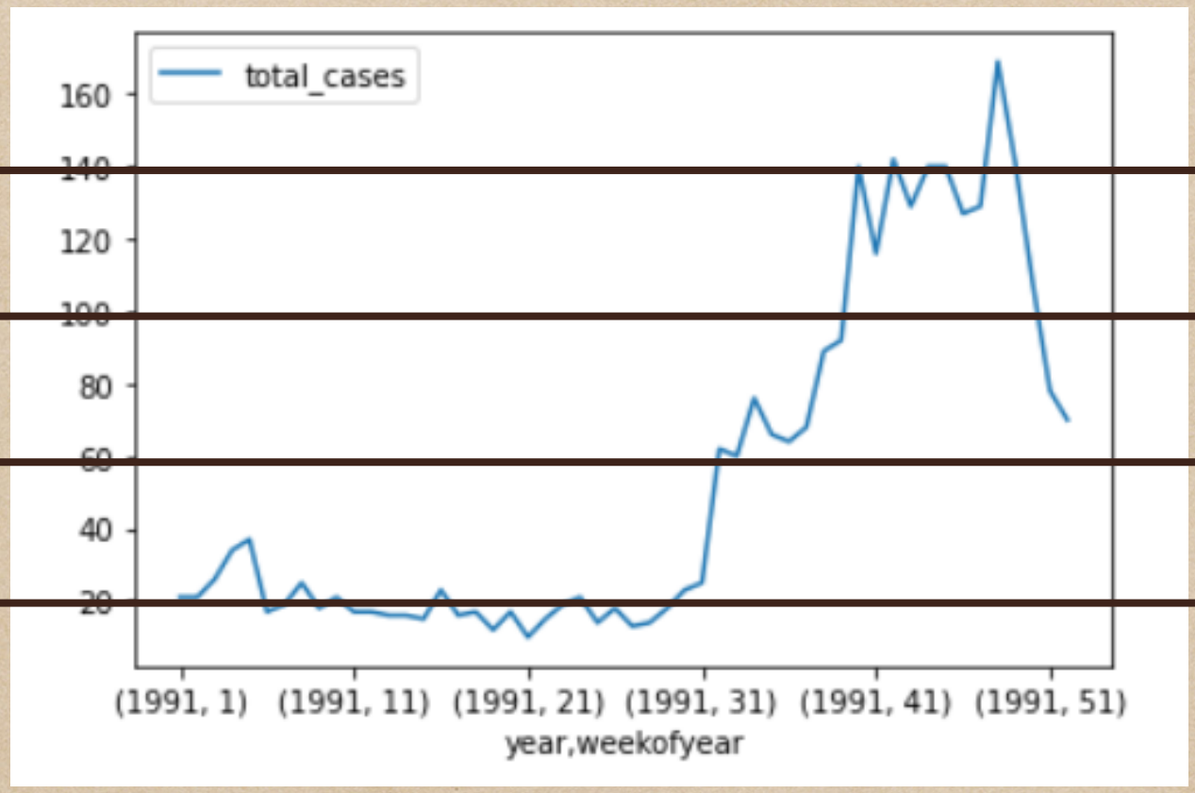
Approach 2

- Feature engineering to model temporal nature
  - E.g., Summary statistics of previous time points
    - See: https://machinelearningmastery.com/basic-feature-engineering-time-series-data-python/
    - Be creative in designing features - has significant impact on final score/rank
- Build regression model

Approach 3

- 2 step modeling approach: classification + regression
  - Classification to first categorise ranges of *y* variable
    - Choose categories manually
    - Should have reasonable explanation
- Build separate regression model for each range
- During prediction: first predict range, then exact value

Example



Ranges:

1. > 140
2. 101 - 140
3. 61 - 100
4. 21 -60
5. < 21

This is just an example for what classes based on target variable range look like. Your choice of both range boundaries and number of ranges should be based on descriptive statistics of the target variable

Approach 3

- 2 step modelling approach: classification + regression
  - Classification to first categorise ranges of *y* variable
    - Choose categories manually
    - Should have reasonable explanation
- Build separate regression model for each range
- During prediction: first predict range, then exact value

Approach 4

- 2 step modelling approach: clustering + regression
  - Cluster data
  - Regression model for each cluster
  - Final prediction: ensemble of regression models

Approach 5

- Use panel data regression
- Econometric approach - extends linear regression concepts
- See:
  - https://towardsdatascience.com/panel-data-regression-a-powerful-time-series-modeling-technique-7509ce043fa8

    [Good concise explanation, but code is in R]

  - https://medium.com/pew-research-center-decoded/using-fixed-and-random-effects-models-for-panel-data-in-python-a795865736ab

    [Same concepts, code in python, statsmodels]

Note

- Feature selection is expected to be beneficial in all 5 approaches
- In approach 3 and 4, you may or may not use features that model temporality (from approach 2)

Rules (1)

- 2-member teams must implement & document at least 3 of these approaches
- 3-member teams must implement & document at least these 5 approaches
- You are welcome to try other approaches in addition to the five outlined

Rules (2)

- For each approach:
  - Submit an entry - predict on the given test data - record leaderboard position
- Each 2-member team must have at least 3 submissions (at least 5 submissions for 3-member teams)
- Leaderboard position screenshots must be submitted

Rules (3)

- At least one of the submitted models:
  - Must be a neural network implemented using PyTorch
  - Must use automated hyperparameter tuning
- They both can be in different or same models/approaches

Rules (4)

- It is important to explain each step you perform (in preprocessing, feature engineering, model training…).
- Ask why you are performing each step and write the reason
- Explain what inference you draw or what you observe after each step (e.g., from descriptive statistics)
- Your submission should be readable like a report

Rules (5)

- You can use any online resource including code: cite them
- No restrictions on choice of classifier, regression or clustering model, except what is stated earlier
- Your explanation should be in *your* own words

Submission

- Per group:
    1. One (or more) Jupyter Notebook(s)
    2. Also submit the same notebook(s) converted to HTML
    3. Document (filename: main, format: ppt/word/pdf…) containing
        - Names and IDs of all group members
        - Leaderboard screenshots
        - [f there are multiple notebooks] sequence in which to view files
    - Upload 1 zipped folder containing all files
- Deadline: Nov 11, 2020 (11:59 AM)

Marks

- Total 40 marks
  - 2-member groups: 3 different solution approaches
    - 3 x 10 = 30 marks
  - 3-member groups: 5 different solution approaches
    - 5 x 6 = 30 marks
  - Explanation (for all teams)
    - 10 marks
- Bonus points for high leaderboard rank (< 500)

TODO...

- Register your group for the competition
- Download the data & explore/understand it

Regular Consultations

- Weekly Consultation Slots
  - Mon 10 am, Thur 3 pm, Fri 11 am (with waiting rooms, 1 hr each)
  - Class hours - after tutorial exercises
  - Encourage each group to consult at least once before project submission
    - You can show us *your* solution to determine if it is correct and if the explanation is sufficient
- Email any of us and schedule for any other time if required