École nationale
de la statistique
et de l'administration
économique

université PARIS-SACLAY

Next Wave Challenge 2019: My Approach

Presentation: France final

EY

Ulrich GOUE

ENS Paris Saclay (MVA)

&

Ensae ParisTech (3A-DS)

19. Mai 2019

Plan

stylized facts

- Many short trips: More than **60%** of the trips last less than 1 minute 20 seconds

stylized facts

- Many short trips: More than **60%** of the trips last less than 1 minute 20 seconds
- Strong stationarity: people tend to remain in the region of interest if they trip start in it and conversely:

stylized facts

- Many short trips: More than **60%** of the trips last less than 1 minute 20 seconds
- Strong stationarity: people tend to remain in the region of interest if they trip start in it and conversely:
- speed measurements seem inconsistent with covered distance:

stylized facts

- Many short trips: More than **60%** of the trips last less than 1 minute 20 seconds
- Strong stationarity: people tend to remain in the region of interest if they trip start in it and conversely:
- speed measurements seem inconsistent with covered distance:
  - **fact**: trips that covered roughly the same distance are associated with a wide range of speeds

stylized facts

- Many short trips: More than **60%** of the trips last less than 1 minute 20 seconds
- Strong stationarity: people tend to remain in the region of interest if they trip start in it and conversely:
- speed measurements seem inconsistent with covered distance:
  - **fact**: trips that covered roughly the same distance are associated with a wide range of speeds
  - **Explanation**: Trips or people movement are constrained by natural barriers that are overlooked by 2D or 3D coordinates.

stylized facts

- Many short trips: More than **60%** of the trips last less than 1 minute 20 seconds
- Strong stationarity: people tend to remain in the region of interest if they trip start in it and conversely:
- speed measurements seem inconsistent with covered distance:
  - **fact**: trips that covered roughly the same distance are associated with a wide range of speeds
  - **Explanation**: Trips or people movement are constrained by natural barriers that are overlooked by 2D or 3D coordinates.
- significant autocorrelation with our target variable and its past up to lag 4 (F-score always greater than 0.68)

stylized facts

- Many short trips: More than **60%** of the trips last less than 1 minute 20 seconds
- Strong stationarity: people tend to remain in the region of interest if they trip start in it and conversely:
- speed measurements seem inconsistent with covered distance:
  - **fact**: trips that covered roughly the same distance are associated with a wide range of speeds
  - **Explanation**: Trips or people movement are constrained by natural barriers that are overlooked by 2D or 3D coordinates.
- significant autocorrelation with our target variable and its past up to lag 4 (F-score always greater than 0.68)
- our target average by hour tends to confirm the existence of **specific hour effect**

Feature Engineering (1)

stylized facts

Based on what has been said in the *stylized facts* we proceeded as follows in the feature engineering.

- **initial speed measurements**:

stylized facts

Based on what has been said in the *stylized facts* we proceeded as follows in the feature engineering.

- **initial speed measurements**:
  - we drop these variables since we couldn't find relevant data to account for the natural barrier effect mentioned above.

stylized facts

Based on what has been said in the *stylized facts* we proceeded as follows in the feature engineering.

- **initial speed measurements**:
    - we drop these variables since we couldn't find relevant data to account for the natural barrier effect mentioned above.
    - we initially try to impute these features by regression but it didn't work very well.

stylized facts

Based on what has been said in the *stylized facts* we proceeded as follows in the feature engineering.

- **initial speed measurements**:
  - we drop these variables since we couldn't find relevant data to account for the natural barrier effect mentioned above.
  - we initially try to impute these features by regression but it didn't work very well.
- **time**:

stylized facts

Based on what has been said in the *stylized facts* we proceeded as follows in the feature engineering.

- **initial speed measurements**:
  - we drop these variables since we couldn't find relevant data to account for the natural barrier effect mentioned above.
  - we initially try to impute these features by regression but it didn't work very well.
- **time**:
  - we convert entry and exit time into seconds

stylized facts

Based on what has been said in the *stylized facts* we proceeded as follows in the feature engineering.

- **initial speed measurements**:
  - we drop these variables since we couldn't find relevant data to account for the natural barrier effect mentioned above.
  - we initially try to impute these features by regression but it didn't work very well.
- **time**:
  - we convert entry and exit time into seconds
  - we compute *trip duration* in seconds

stylized facts

Based on what has been said in the *stylized facts* we proceeded as follows in the feature engineering.

- **initial speed measurements**:
  - we drop these variables since we couldn't find relevant data to account for the natural barrier effect mentioned above.
  - we initially try to impute these features by regression but it didn't work very well.
- **time**:
  - we convert entry and exit time into seconds
  - we compute *trip duration* in seconds
  - we also extract trip departure hour and trip arrival.

Feature Engineering (2)

- **proximity**:

⟹ Notice that the so-called forward variables should be lagged to prevent data leakage

Feature Engineering (2)

- **proximity**:
  - create dummy variable capturing if trip starts in the region of interest (ROI, hereafter)

$\Longrightarrow$ Notice that the so-called forward variables should be lagged to prevent data leakage

Feature Engineering (2)

- **proximity**:
  - create dummy variable capturing if trip starts in the region of interest (ROI, hereafter)
  - create dummy variable to capture *strange trip* of duration zero where the people are (*approximately*) thus stuck at the same place.

$\Longrightarrow$ Notice that the so-called forward variables should be lagged to prevent data leakage

Feature Engineering (2)

- **proximity**:
  - create dummy variable capturing if trip starts in the region of interest (ROI, hereafter)
  - create dummy variable to capture *strange trip* of duration zero where the people are (*approximately*) thus stuck at the same place.
  - compute the distance between the initial coordinates and the center of the region of interest (both L1 and L2)

$\Longrightarrow$ Notice that the so-called forward variables should be lagged to prevent data leakage

Feature Engineering (2)

- **proximity**:
  - create dummy variable capturing if trip starts in the region of interest (ROI, hereafter)
  - create dummy variable to capture *strange trip* of duration zero where the people are (*approximately*) thus stuck at the same place.
  - compute the distance between the initial coordinates and the center of the region of interest (both L1 and L2)
- **trip dynamics (forward variables)**:

$\Longrightarrow$ Notice that the so-called forward variables should be lagged to prevent data leakage

Feature Engineering (2)

- **proximity**:
  - create dummy variable capturing if trip starts in the region of interest (ROI, hereafter)
  - create dummy variable to capture *strange trip* of duration zero where the people are (*approximately*) thus stuck at the same place.
  - compute the distance between the initial coordinates and the center of the region of interest (both L1 and L2)
- **trip dynamics (forward variables)**:
- compute the same distance as above but with the arrival coordinates

$\Longrightarrow$ Notice that the so-called forward variables should be lagged to prevent data leakage

Feature Engineering (2)

- **proximity**:
  - create dummy variable capturing if trip starts in the region of interest (ROI, hereafter)
  - create dummy variable to capture *strange trip* of duration zero where the people are (*approximately*) thus stuck at the same place.
  - compute the distance between the initial coordinates and the center of the region of interest (both L1 and L2)
- **trip dynamics (forward variables)**:
- compute the same distance as above but with the arrival coordinates
- compute associate *synthetic* speed based on both distances.

$\Longrightarrow$ Notice that the so-called forward variables should be lagged to prevent data leakage

Feature Engineering (3)

- **Region encoding**: it is based on L1 distance.

Feature Engineering (3)

- **Region encoding**: it is based on L1 distance.
  - If the limits of the ROI are given by $x_{min}, x_{max}, y_{min}, y_{max}$

Feature Engineering (3)

- **Region encoding**: it is based on L1 distance.
  - If the limits of the ROI are given by $x_{min}, x_{max}, y_{min}, y_{max}$
  - A 2D point (x, y) is encoded by $(x*, y*)$ as follows:

$$(w*, y*) = \arg\min \mid x - x' \mid + \mid y - y' \mid \text{ s.c.}$$

with $x_{min} \leq x' \leq x_{max}, y_{min} \leq y' \leq y_{max}$

Feature Engineering (3)

- **Region encoding**: it is based on L1 distance.
  - If the limits of the ROI are given by $x_{min}, x_{max}, y_{min}, y_{max}$
  - A 2D point (x, y) is encoded by $(x*, y*)$ as follows:

    $$(w*, y*) = \arg\min \mid x - x' \mid + \mid y - y' \mid \text{ s.c.}$$

    with $x_{min} \leq x' \leq x_{max}, y_{min} \leq y' \leq y_{max}$
  - these features are computed both for arrival and departure coordinates. *Again those features related to arrival coordinates should be lagged.*

Feature Engineering (3)

- **Region encoding**: it is based on L1 distance.
  - If the limits of the ROI are given by $x_{min}, x_{max}, y_{min}, y_{max}$
  - A 2D point (x, y) is encoded by $(x*, y*)$ as follows:

    $$(w*, y*) = \arg\min \mid x - x' \mid + \mid y - y' \mid \text{ s.c.}$$

    with $x_{min} \leq x' \leq x_{max}, y_{min} \leq y' \leq y_{max}$
  - these features are computed both for arrival and departure coordinates. *Again those features related to arrival coordinates should be lagged.*
- **data processing**:

Feature Engineering (3)

- **Region encoding**: it is based on L1 distance.
  - If the limits of the ROI are given by $x_{min}, x_{max}, y_{min}, y_{max}$
  - A 2D point (x, y) is encoded by $(x*, y*)$ as follows:

$$(w*, y*) = \arg\min \mid x - x' \mid + \mid y - y' \mid \text{ s.c.}$$

    with $x_{min} \leq x' \leq x_{max}, y_{min} \leq y' \leq y_{max}$
  - these features are computed both for arrival and departure coordinates. *Again those features related to arrival coordinates should be lagged.*
- **data processing**:
  - we downloaded weather data that we finally didn't use.

Feature Engineering (3)

- **Region encoding**: it is based on L1 distance.
  - If the limits of the ROI are given by $x_{min}, x_{max}, y_{min}, y_{max}$
  - A 2D point (x, y) is encoded by $(x*, y*)$ as follows:

$$(w*, y*) = \arg\min \mid x - x' \mid + \mid y - y' \mid \text{ s.c.}$$

    with $x_{min} \leq x' \leq x_{max}, y_{min} \leq y' \leq y_{max}$
  - these features are computed both for arrival and departure coordinates. *Again those features related to arrival coordinates should be lagged.*
- **data processing**:
  - we downloaded weather data that we finally didn't use.
  - The 2D coordinates are divided by 10^6 to narrow the variation range of coordinates

Feature Engineering (3)

- **Region encoding**: it is based on L1 distance.
  - If the limits of the ROI are given by $x_{min}, x_{max}, y_{min}, y_{max}$
  - A 2D point (x, y) is encoded by $(x*, y*)$ as follows:

$$(w*, y*) = \arg\min \mid x - x' \mid + \mid y - y' \mid \text{ s.c.}$$

    with $x_{min} \leq x' \leq x_{max}, y_{min} \leq y' \leq y_{max}$
  - these features are computed both for arrival and departure coordinates. *Again those features related to arrival coordinates should be lagged*.
- **data processing**:
  - we downloaded weather data that we finally didn't use.
  - The 2D coordinates are divided by 10^6 to narrow the variation range of coordinates
  - the file `makedata.py` performs the feature engineering while `getfeatures.py` adds the lags and pads the devices recording few trips for our RNN.

Models without lags

Number of features = 24

Earlier experiments (1/2)

- **MLP Regression** with one hidden layer of 10 neurons: we train a simple MLP to predict the arrival coordinates and test if the prediction was in the ROI. It scored aroud ≈ 0.76, and the MSE $\approx 8.10^{-4}$

Models without lags

Number of features = 24

Earlier experiments (1/2)

- **MLP Regression** with one hidden layer of 10 neurons: we train a simple MLP to predict the arrival coordinates and test if the prediction was in the ROI. It scored aroud ≈ 0.76, and the MSE $\approx 8.10^{-4}$
- **MLP Regression + Tree**: It scored ≈ 0.85

Models without lags

Number of features = 24

Earlier experiments (1/2)

- **MLP Regression** with one hidden layer of 10 neurons: we train a simple MLP to predict the arrival coordinates and test if the prediction was in the ROI. It scored aroud ≈ 0.76, and the MSE $\approx 8.10^{-4}$
- **MLP Regression + Tree**: It scored ≈ 0.85
  - *Intuituion 1*: the target is generated by the tests $x_{min} \leq x' \leq x_{max}, y_{min} \leq y' \leq y_{max}$ is tree of depth 4

Models without lags

Number of features = 24

Earlier experiments (1/2)

- **MLP Regression** with one hidden layer of 10 neurons: we train a simple MLP to predict the arrival coordinates and test if the prediction was in the ROI. It scored aroud ≈ 0.76, and the MSE $\approx 8.10^{-4}$
- **MLP Regression + Tree**: It scored ≈ 0.85
  - *Intuituion 1*: the target is generated by the tests $x_{min} \leq x' \leq x_{max}, y_{min} \leq y' \leq y_{max}$ is tree of depth 4
  - *Intuition 2*: using the tree in the second step will correct at some extent some systematic errors madse by the first step MLP regressor.

Model with lags (up to 4)

Number of features $= 120$

Earlier experiments (2/2)

- **LSTM Regression**: It scored ≈ 0.848

Model with lags (up to 4)

Number of features $= 120$

Earlier experiments (2/2)

- **LSTM Regression**: It scored ≈ 0.848
- **LSTM classification**: It scored ≈ 0.884

Model with lags (up to 4)

Number of features = 120

Earlier experiments (2/2)

- **LSTM Regression**: It scored ≈ 0.848
- **LSTM classification**: It scored ≈ 0.884
- **Tree**: It scored ≈ 0.882

best regression

- Here we try to address the observation made in the beginning, since every point of the plane cannot be predicted as an arrival point

best regression

- Here we try to address the observation made in the beginning, since every point of the plane cannot be predicted as an arrival point
- We first estimate the most likely arrival points by the **mean shift algorithm**. We got 1632 potential modes that we called anchor points

best regression

- Here we try to address the observation made in the beginning, since every point of the plane cannot be predicted as an arrival point
- We first estimate the most likely arrival points by the **mean shift algorithm**. We got 1632 potential modes that we called anchor points
- We use an LSTM architecture to predict the probability of a trip to end near our anchors.

best regression

- Here we try to address the observation made in the beginning, since every point of the plane cannot be predicted as an arrival point
- We first estimate the most likely arrival points by the **mean shift algorithm**. We got 1632 potential modes that we called anchor points
- We use an LSTM architecture to predict the probability of a trip to end near our anchors.
- The final prediction is then the weighted mean of the anchors with estimated probability as weights.

best regression

- Here we try to address the observation made in the beginning, since every point of the plane cannot be predicted as an arrival point
- We first estimate the most likely arrival points by the **mean shift algorithm**. We got 1632 potential modes that we called anchor points
- We use an LSTM architecture to predict the probability of a trip to end near our anchors.
- The final prediction is then the weighted mean of the anchors with estimated probability as weights.
- If we reduce MSE by 50% from $\approx 8.10^{-4}$ to $\approx 3.5 \times 10^{-4}$, the F1-score remains ≈ 0.865

best classification

Best Classification

- This is a slight modification of the LSTM presented above, except that we graft another network to learn specific embeddings for all the categorical variables

best classification

Best Classification

- This is a slight modification of the LSTM presented above, except that we graft another network to learn specific embeddings for all the categorical variables
- A part of the network learn the embeddings, another processed the time series through LSTM and merge their outputs to generate the final predictions

best classification

Best Classification

- This is a slight modification of the LSTM presented above, except that we graft another network to learn specific embeddings for all the categorical variables
- A part of the network learn the embeddings, another processed the time series through LSTM and merge their outputs to generate the final predictions
- It scored ≈ 0.894

Final thoughts

- In this challenge we were left a little bit frustrated. Many other approaches that we attempted didn't better our performances.

Final thoughts

- In this challenge we were left a little bit frustrated. Many other approaches that we attempted didn't better our performances.
  - **external data**: weather data didn't improve our performances

Final thoughts

- In this challenge we were left a little bit frustrated. Many other approaches that we attempted didn't better our performances.
  - **external data**: weather data didn't improve our performances
  - **multitask learning**: we tried to simultaneously predict the arrival coordinates and the label

Final thoughts

- In this challenge we were left a little bit frustrated. Many other approaches that we attempted didn't better our performances.
  - **external data**: weather data didn't improve our performances
  - **multitask learning**: we tried to simultaneously predict the arrival coordinates and the label
  - **data augmenting**: it was also not fruitful (maybe explore GAN augmenting methods)

Final thoughts

- In this challenge we were left a little bit frustrated. Many other approaches that we attempted didn't better our performances.
  - **external data**: weather data didn't improve our performances
  - **multitask learning**: we tried to simultaneously predict the arrival coordinates and the label
  - **data augmenting**: it was also not fruitful (maybe explore GAN augmenting methods)
- The main failure of our approach is that it didn't cope well with the two kinds of trips that we observed (short and long)

Final thoughts

- In this challenge we were left a little bit frustrated. Many other approaches that we attempted didn't better our performances.
  - **external data**: weather data didn't improve our performances
  - **multitask learning**: we tried to simultaneously predict the arrival coordinates and the label
  - **data augmenting**: it was also not fruitful (maybe explore GAN augmenting methods)
- The main failure of our approach is that it didn't cope well with the two kinds of trips that we observed (short and long)
  - speed data are obviously more important for long trips. Unfortunately we dropped it because we didn't know how to properly impute them.

Final thoughts

- In this challenge we were left a little bit frustrated. Many other approaches that we attempted didn't better our performances.
  - **external data**: weather data didn't improve our performances
  - **multitask learning**: we tried to simultaneously predict the arrival coordinates and the label
  - **data augmenting**: it was also not fruitful (maybe explore GAN augmenting methods)
- The main failure of our approach is that it didn't cope well with the two kinds of trips that we observed (short and long)
  - speed data are obviously more important for long trips. Unfortunately we dropped it because we didn't know how to properly impute them.
  - When we tried a specific model for long trips, we lost power since we surrender 66% of the sample while neural networks require large samples to learn.

THANK YOU FOR YOUR ATTENTION !!!