AXA Data Challenge 2016

Yellow

BELLEC Maxime, HUNG Chia-Man, LIU Zhengying

Master Data Science

January 18, 2017

Summary

1. Data pre-processing & Feature engineering
2. Data visualization & Preliminary analysis
3. Our approaches
   - A first simple approach
   - A generalized version: Linear LinEx Regression
   - Random Forest
4. Conclusion

Summary

1. Data pre-processing & Feature engineering
2. Data visualization & Preliminary analysis
3. Our approaches
   - A first simple approach
   - A generalized version: Linear LinEx Regression
   - Random Forest
4. Conclusion

Data pre-processing

- Only a few columns of the training data are in test data
  ⇒ Keep only 3 columns: `DATE`, `ASS_ASSIGNMENT` and `CSPL_RECEIVED_CALLS`
- Multiple columns for each (`DATE`, `ASS_ASSIGNMENT`)
  ⇒ sum the `CSPL_RECEIVED_CALLS`

Sum the repeated rows

	ASS_ASSIGNMENT	DATE	CSPL_RECEIVED_CALLS
9696073	Téléphonie	2013-11-18 11:30:00	53
9696307	Téléphonie	2013-11-18 11:30:00	72
9696347	Téléphonie	2013-11-18 11:30:00	24
9696369	Téléphonie	2013-11-18 11:30:00	28
9696372	Téléphonie	2013-11-18 11:30:00	53
9696375	Téléphonie	2013-11-18 11:30:00	33
9696392	Téléphonie	2013-11-18 11:30:00	24
9696404	Téléphonie	2013-11-18 11:30:00	135
9696415	Téléphonie	2013-11-18 11:30:00	26
9696441	Téléphonie	2013-11-18 11:30:00	40
9696444	Téléphonie	2013-11-18 11:30:00	53
9696527	Téléphonie	2013-11-18 11:30:00	92

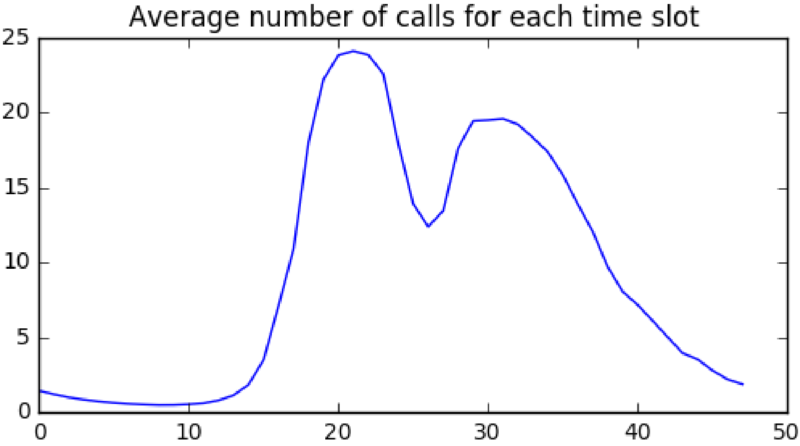
Feature engineering

	DATE	ASS_ASSIGNMENT	CSPL_RECEIVED_CALLS	slot	dayofweek	month	year	day_off	day_after_day_off
0	2011-01-01	Crises	0	0	5	1	2011	True	False
1	2011-01-01	Domicile	0	0	5	1	2011	True	False
2	2011-01-01	Gestion	0	0	5	1	2011	True	False
3	2011-01-01	Gestion - Accueil Telephonique	0	0	5	1	2011	True	False
4	2011-01-01	Gestion Assurances	0	0	5	1	2011	True	False

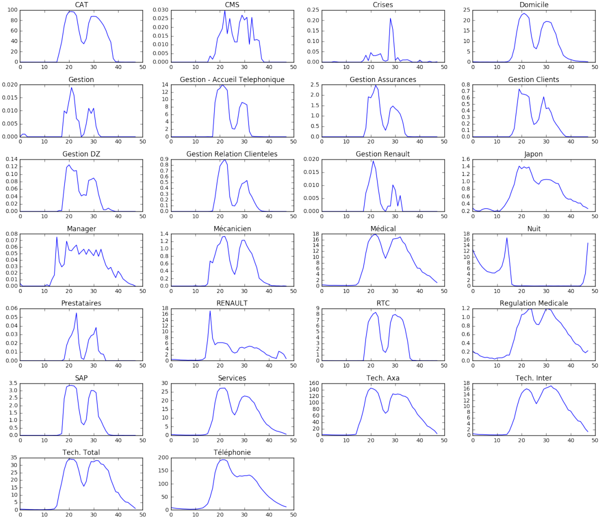
Summary

1. Data pre-processing & Feature engineering
2. Data visualization & Preliminary analysis
3. Our approaches
   - A first simple approach
   - A generalized version: Linear LinEx Regression
   - Random Forest
4. Conclusion

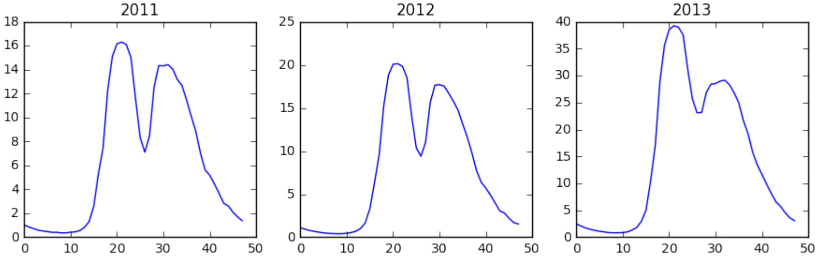
Data visualization



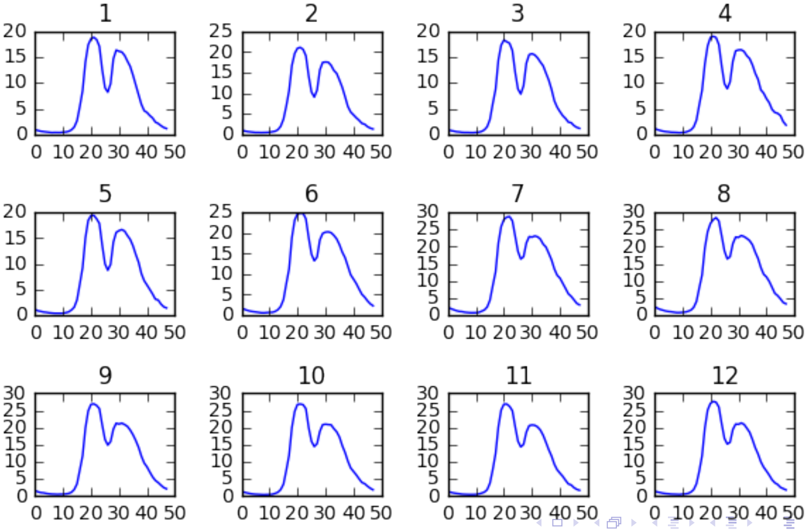
Data visualization - ASS_ASSIGNMENT



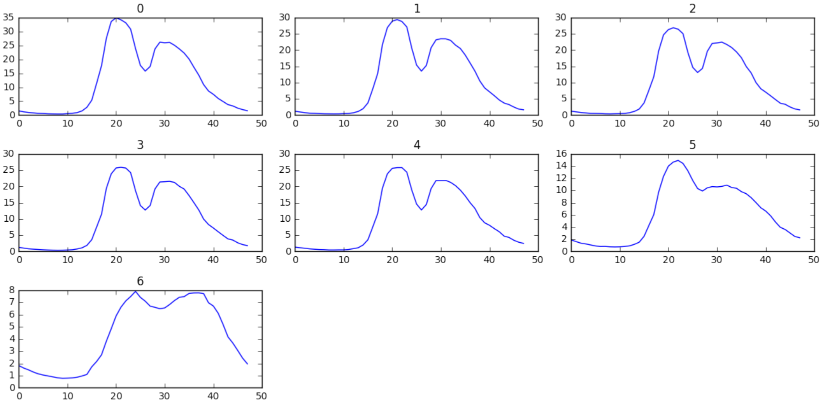
Data visualization - Year



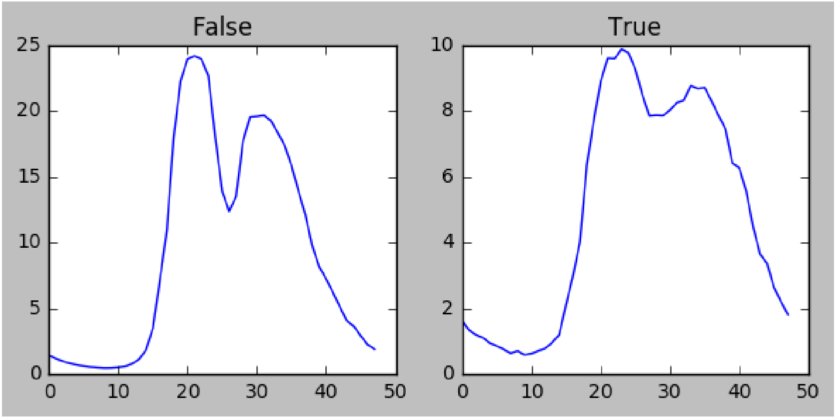
Data visualization - Month



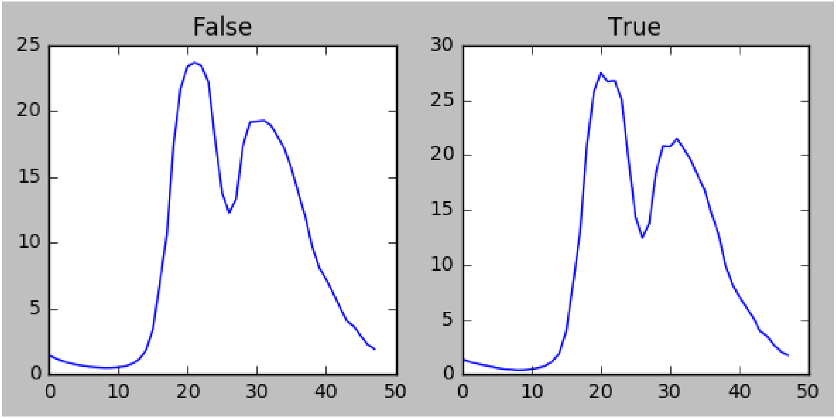
Data visualization - Weekday



Data visualization - DAY_OFF



Data visualization - DAY_AFTER_DAY_OFF



Summary

1. Data pre-processing & Feature engineering
2. Data visualization & Preliminary analysis
3. Our approaches
   - A first simple approach
   - A generalized version: Linear LinEx Regression
   - Random Forest
4. Conclusion

1st approach - simple approach

- For a given `ASS_ASSIGNMENT` and weekly time slot, such as Tuesday 09:00-09:30, the `CSPL_RECEIVED_CALLS` are more or less stationary
- The idea is to predict for each the best stationary value by minimizing the empirical loss

1st approach - simple approach

The empirical loss function

$$R(\hat{y}) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \hat{y})$$

1st approach - simple approach

The empirical loss function

$$R(\hat{y}) = \frac{1}{n}\sum_{i=1}^{n}\ell(y_i, \hat{y})$$

The best constant value

$$R'(\hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(-\alpha e^{\alpha(y_i-\hat{y})} + \alpha) = 0$$

$$\Longrightarrow \frac{1}{n}\sum_{i=1}^{n} e^{\alpha(y_i-\hat{y})} = 1 \tag{1}$$

$$\Longrightarrow \hat{y} = \log\frac{1}{n}\sum_{i=1}^{n} e^{\alpha y_i} = softmax(\alpha Y) - \log n$$

1st approach - simple approach

Advantages

- Simple model
- Explainable
- Use the real loss function LinEx

Disadvantages

- Too many parameters (about 10000 lines in `predict_table`)
- Many of these parameters are correlated

Result

2.55 on the leader board

2nd approach - a generalized version

- Our 1st approach can be regarded as a **linear regression** model
  where the feature vector is one-hot encoding for all possible (ASS_ASSIGNMENT, slot, dayofweek) tuples.
- We extend it by considering more features (month, day_off)
- More general: consider **all possible combinations of all features!**

2nd approach - a generalized version

- Our 1st approach can be regarded as a **linear regression** model
  where the feature vector is one-hot encoding for all possible (ASS_ASSIGNMENT, slot, dayofweek) tuples.
- We extend it by considering more features (month, day_off)
- More general: consider **all possible combinations of all features!**
  $\Rightarrow$ Linear LinEx Regression on Combined Features

Combined Feature matrix

- For the row
  `(ASS_ASSIGNMENT, dayofweek, month, slot)`
  `= (Crises, 5, 1, 0)`
- We associate a vector of 0 and 1s, where the 1s are in columns corresponding to

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1, slot=0)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1, slot=0)
- (dayofweek=5, month=1, slot=0)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1, slot=0)
- (dayofweek=5, month=1, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1, slot=0)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1, slot=0)
- (dayofweek=5, month=1, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1, slot=0)
- (ASS_ASSIGNMENT=Crises, dayofweek=5, slot=0)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1, slot=0)
- (dayofweek=5, month=1, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1, slot=0)
- (ASS_ASSIGNMENT=Crises, dayofweek=5, slot=0)
- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1, slot=0)
- (dayofweek=5, month=1, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1, slot=0)
- (ASS_ASSIGNMENT=Crises, dayofweek=5, slot=0)
- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1)
- (month=1, slot=0)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1, slot=0)
- (dayofweek=5, month=1, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1, slot=0)
- (ASS_ASSIGNMENT=Crises, dayofweek=5, slot=0)
- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1)
- (month=1, slot=0)
- (dayofweek=5, slot=0)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1, slot=0)
- (dayofweek=5, month=1, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1, slot=0)
- (ASS_ASSIGNMENT=Crises, dayofweek=5, slot=0)
- (ASS_ASSIGNMENT=Crises, dayofweek=5, month=1)
- (month=1, slot=0)
- (dayofweek=5, slot=0)
- (dayofweek=5, month=1)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, slot=0)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1)
- (ASS_ASSIGNMENT=Crises, dayofweek=5)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1)
- (ASS_ASSIGNMENT=Crises, dayofweek=5)
- (slot=0)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1)
- (ASS_ASSIGNMENT=Crises, dayofweek=5)
- (slot=0)
- (month=1)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1)
- (ASS_ASSIGNMENT=Crises, dayofweek=5)
- (slot=0)
- (month=1)
- (dayofweek=5)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1)
- (ASS_ASSIGNMENT=Crises, dayofweek=5)
- (slot=0)
- (month=1)
- (dayofweek=5)
- (ASS_ASSIGNMENT=Crises)

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1)
- (ASS_ASSIGNMENT=Crises, dayofweek=5)
- (slot=0)
- (month=1)
- (dayofweek=5)
- (ASS_ASSIGNMENT=Crises)
- ()

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1)
- (ASS_ASSIGNMENT=Crises, dayofweek=5)
- (slot=0)
- (month=1)
- (dayofweek=5)
- (ASS_ASSIGNMENT=Crises)
- ()

(ASS_ASSIGNMENT, dayofweek, month, slot) = (Crises, 5, 1, 0)

- (ASS_ASSIGNMENT=Crises, slot=0)
- (ASS_ASSIGNMENT=Crises, month=1)
- (ASS_ASSIGNMENT=Crises, dayofweek=5)
- (slot=0)
- (month=1)
- (dayofweek=5)
- (ASS_ASSIGNMENT=Crises)
- ()
  $\Uparrow$ intercept

Combined Feature matrix

- A feature matrix of shape (1030829,147784)!
- But each row has only **16 non-zero terms**
- $\Rightarrow$ Use `scipy.sparse.csr_matrix`

Linear linex regression

The loss function

$$\frac{1}{n}\sum_{i=1}^{n}\ell(y_i, x_i^\top\theta) + \frac{\lambda}{2}\|\theta\|_2^2$$

where $\ell(x, y) = LinEx(x, y) = \exp(\alpha(x - y)) - \alpha(x - y) - 1$ (LinEx regression)

Learning algorithm

SVRG (Stochastic Variance Reduced Gradient) algorithm

SVRG algorithm

Input: starting point θ_0, learning rate $\eta > 0$
Put $\tilde{\theta}^1 \leftarrow \theta_0$
For $k = 1, 2, \ldots$ until convergence do

1. Put $\theta_0^k \leftarrow \tilde{\theta}_0^1$
2. Compute $\mu = \nabla f(\tilde{\theta}^k)$
3. For $t = 0, \ldots, m-1$:
   - Pick uniformly at random i in $\{1, \ldots, n\}$
   - Apply the step

$$\theta_{t+1}^k \leftarrow \theta_t^k - \eta(\nabla f_i(\theta_t^k) - \nabla f_i(\tilde{\theta}^k) + \mu)$$

Set

$$\tilde{\theta}^k \leftarrow \frac{1}{m} \sum_{t=1}^{m} \theta_t^k$$

Return last θ_t^k

2nd approach - Linear LinEx Regression

Advantages

- The loss function is convex
- Very general, containing many approaches as special case
- Explainable
- Use the real loss function LinEx

Disadvantages

- Many parameters (about 150000 of them)
- Hard to optimize

Result

1.99 on the leader board

3rd approach - Random Forest

- Use all the features in feature engineering
- No categorical values in sklearn $\Rightarrow$ one-hot encoding
- Remove Evenements and Gestion Amex
- Cross validation (80% training, 20% testing)
- Multiply by $C = 2.4$

3rd approach - Random Forest

Advantages

- Robust model
- Existing library
- Relatively good results

Disadvantages

- Parameter tuning
- Hard to use a custom loss function

Result

1.175 on the leaderboard

Summary

1. Data pre-processing & Feature engineering
2. Data visualization & Preliminary analysis
3. Our approaches
   - A first simple approach
   - A generalized version: Linear LinEx Regression
   - Random Forest
4. Conclusion

Conclusion

- For prediction, when collecting data on past time, make sure this data will also be available for future times, otherwise they are not useful features for prediction
- A lot of features can be created on DATE and it can be enough when the data actually mostly depends on DATE