Modèle de prédiction pour des series de données temporelles

Mémoire de certification

Conduire un projet de sciences de données CNCP 1527

Data Science Starter Program DSSP 6 – 2017

Alaoui Mohamed avec JetPack Data

Plan

- JetPack Data et expression du besoin
- Travaux effectués, démarches et expérimentations
- Ouvertures

JetPack Data

- CEO Shankar Arul – French Tech 2017, Incubé chez Numa à StationF
- Exploration intelligentes de data
- Go to site :
  - https://www.jetpackdata.com/landing

JetPack Data - Architecture

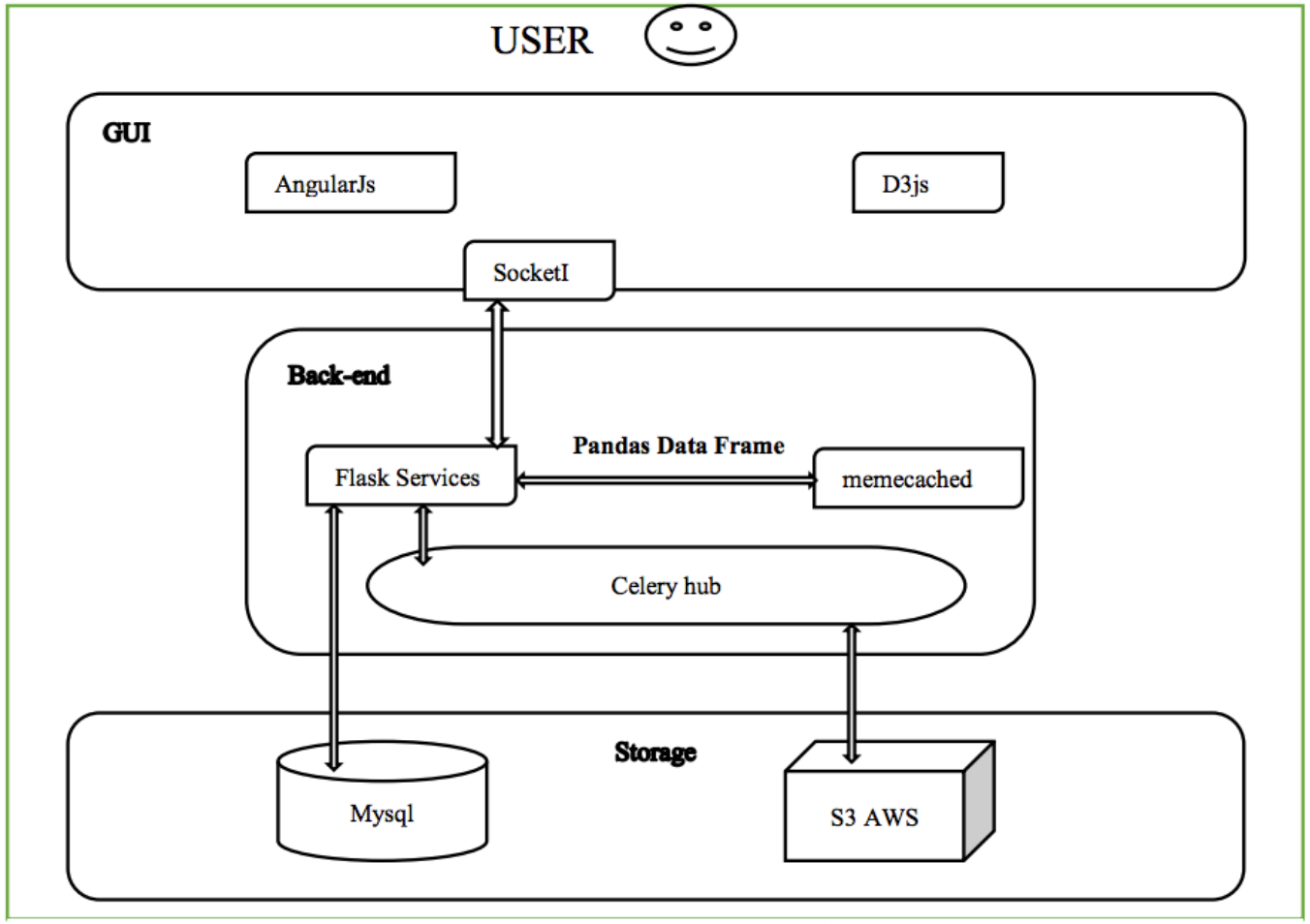


Figure 3 Architecture logicielle

Besoin JetPack Data

- API simple et robuste permettant l' intégration facile de différents models
- Lib python facile à utiliser
- Intégration facile avec la plateforme d'exploration de données
- Facilité d' installation

Travaux

- Driver de la démarche :
  - Répondre au besoin de JetPack Data
  - Valider la certification DSSP6

- Expérimentation :
  - Modèle statistique :
    - ARIMA & Prophet
  - Pipeline de forecasting

- Machine Learning – RNN LSTM

Modèle statistiques

- Description des jeux de data
- Stationnarité
- Arima
- Prophet
- https://github.com/jetpackdata/jpdforcaster/blob/master/notebooks/dataSet-description-rapportDSSP6.ipynb

Pipeline de forecasting

- Exemple d'utilisation :
  - https://github.com/jetpackdata/jpdforcaster
  - https://github.com/jetpackdata/jpdforcaster/blob/master/python/jpddsforcasting/usage.py
- jpddsforcasting
  - https://pypi.python.org

Machine Learning & Times Series

- Adaptation à l'apprentissage supervisé
- RNN
- RNN LSTM
- Résultat :
  - https://github.com/jetpackdata/jpdforcaster/blob/master/notebooks/LSTM-rapportDSSP6.ipynb

Consolidation et critique des résultats obtenus

La mesure de performance choisie est la « Normalized Root Mean Squared Error ». Elle permet une normalisation de l'erreur pour comparer les modèles par types de séries temporelles.

Modèles statistiques :

Afin de pouvoir comparer les modèles, nous allons utiliser une profondeur d'historique en nous basant sur les paramètres optimaux du modèle ARIMA pour la série TS-volume.
Nous allons considérer une profondeur d'historique de 4 pour toutes les séries afin de prédire la 5ième valeur.

NRMSE DataSet/Model	ARIMA ARIMA (4,0,3)	Prophet	SMA (avec tendance et saisonnalite´ de Prophet) Windows Size : 4	EWMA (avec tendance et saisonnalité de Prophet) Windows Size : 4
TS-volume (tendance et saisonnalité évidente)	0.1295	0.0458	0.0881	0.1303
TS-revenu (tendance et saisonnalité non évidente)	0.1459	0.1606	0.1203	0.0981
TS-Stock (stochastique)	0.0901	0.0694	0.0423	0.0369

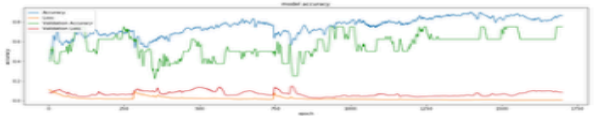
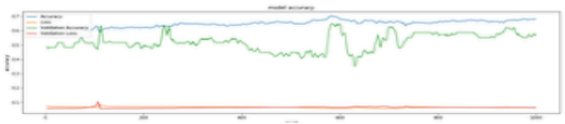
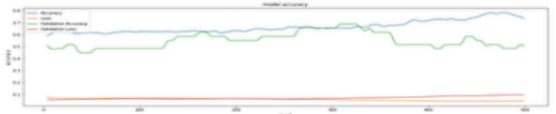
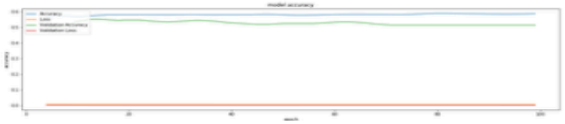
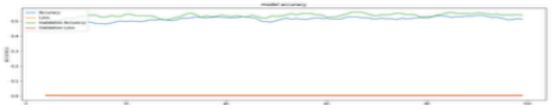
Pour les séries à saisonnalité et tendance évidentes nous remarquons que la librairie Prophet de Facebook possède une performance 2 ou 3 fois meilleure que les autres méthodes. En effet, cette dernière a été conçue pour être robuste dans ces cas précis vu les besoins en analyse de séries temporelles de Facebook.

Par ailleurs, pour les séries de types « revenu économique » ou « stock financier » dont les processus sont plus complexes et aléatoires, les modèles régressifs sont de meilleure performance. En effet, l'historique immédiat joue un rôle important sur ce type de séries quand leur résiduel (hors tendance et saisonnalité) est localement stationnaire, ce qui est le cas de TS- revenu et TS-Stock.

Consolidation et critique des résultats obtenus

Modèles de Deep Learning :

Afin de pouvoir comparer les modèles pour chacune des séries, nous allons utiliser des vecteurs de taille 5 et nous prédirons les 2 dernières valeurs à l'aide des 3 premières.
Synthèses des résultats :

NRMSE DataSet/Model	LSTM Cas 1	LSTM avec Dropout et Régularisation Cas 2 dropout : 0.4 regul : 0.01
TS-volume (tendance et saisonnalité évidente) – 71/8 1700 itérations LSTM 30 nodes hidden layer	0.1424 	0.2550 
TS-revenu (tendance et saisonnalité non évidente) – 259/29 500 itérations LSTM 10 nodes hidden layer	0.2178 	0.1901 
TS-Stock (stochastique) – 901/101 100 itérations LSTM 10 nodes hidden layer	0.1311 	0.1498 

Consolidation et critique des résultats obtenus

Dans le tableau ci-dessus nous avons consolidé les NRMSE des différents jeux de données avec leurs courbes d'apprentissage.
Nous allons considérer dans un premier temps un cas de référence, cas 1, des réseaux de neurones LSTM sans optimisation.

Dans un deuxième temps, cas 2, nous allons contraindre le modèle en utilisant une régularisation puis réduire aléatoirement à chaque itération le nombre de neurones utilisées pour entrainer le modèle. Ceci permettra d'atténuer les effets de sur-apprentissage.
Pour ce faire nous utiliseront un « dropout » à 0.4.

- La courbe bleue correspond à l'évolution de la précision du modèle à chaque itération
- La courbe verte correspond à l'évolution de la précision de la cross-validation
- Les courbes rouges et oranges étant les fonctions de coût à minimiser

Le comportement du modèle sur la série TS-volume (avec saisonnalité et tendance évidente) révèle un taux d'apprentissage ascendant tout au long des 1700 itérations pour les cas 1 et 2.
Le réseau LSTM se comporte aussi bien qu'un modèle statistique si nous augmentons le nombre d'itérations et de neurones. Cependant, l'investissement en ressources de calculs serait plus conséquent, et pour ce type de série, la librairie Prophet fournit de très bons résultats avec moins de ressource de calcul.

Les résultats pour les séries à saisonnalité et tendance cachées ou stochastiques sont plus mitigés.
Les fonctions de coût restent constantes et proches de zéro. Nous sommes face à un cas de sur- apprentissage, probablement dû à la complexité du modèle et surtout à la taille des séries.
Nous constatons par ailleurs une constance de la précision tout au long des itérations dans le cas 1. Dans le cas 2, nous remarquons l'apparition d'une tendance ascendante suite à l'ajout du « dropout » et la « régularisation ». Les optimisations utilisées dans le cas 2 améliorent le modèle, cependant il est nécessaire d'avoir des données plus volumineuses afin de constater clairement cette amélioration et pouvoir profiter de toute la puissance des RNN LSTM.

Commentaires :

L'investissement dans un RNN LSTM est intéressant si nous possédons un jeu d'entrainement conséquent et assez de ressources pour permettre une optimisation des hyper-paramètres et de la structure du réseau.
Pour des jeux de données plus réduits, les modèles statistiques sont plus légers et donnent de bons résultats. La librairie Prophet est robuste pour des séries temporelles avec tendance et saisonnalité ayant un historique d'un an ou plus.
Pour les séries de petite taille de types stochastiques et localement non-stationnaires, les modèles régressifs donnent de bons résultats pour des prédictions à court terme.

Ouvertures

- Investissement sur cette librairie dans le cadre :
  - de ma certification IOT en cours
  - de la startup CULTYDATA créé le 1er novembre 2018